

# 000 001 002 003 004 005 006 007 008 009 010 011 012 DEEPSCHOLAR-BENCH: A LIVE BENCHMARK AND AUTOMATED EVALUATION FOR GENERATIVE RESEARCH SYNTHESIS

013  
014  
015  
016  
017  
018  
019  
020  
021  
022  
023  
024  
025  
026  
027  
028  
029  
030  
031  
032  
033  
034  
035  
036  
037  
038  
039  
040  
041  
042  
043  
044  
045  
046  
047  
048  
049  
050  
051  
052  
053  
**Anonymous authors**  
Paper under double-blind review

## ABSTRACT

The ability to research and synthesize knowledge is central to human expertise and progress. A new class of AI systems—designed for generative research synthesis—aims to automate this process by retrieving information from the live web and producing long-form, cited reports. Yet, evaluating such systems remains an open challenge: existing question-answering benchmarks focus on short, factual answers, while expert-curated datasets risk staleness and data contamination. Neither captures the complexity and evolving nature of real research synthesis tasks. We introduce DeepScholar-bench, a live benchmark and automated evaluation framework for generative research synthesis. DeepScholar-bench draws queries and human-written exemplars from recent, high-quality ArXiv papers and evaluates a real synthesis task: generating a related work section by retrieving, synthesizing, and citing prior work. Our automated framework holistically measures performance across three key dimensions—knowledge synthesis, retrieval quality, and verifiability. To further future work, we also contribute DeepScholar-ref, a simple, open-source reference pipeline, which is implemented on the LOTUS framework and provides a strong baseline. Using DeepScholar-bench, we systematically evaluate prior open-source systems, search agents with strong models, OpenAI’s DeepResearch, and DeepScholar-ref. We find DeepScholar-bench is far from saturated: no system surpasses a geometric mean of 31% across all metrics. These results highlight both the difficulty and importance of DeepScholar-bench as a foundation for advancing AI systems capable of generative research synthesis.

## 1 INTRODUCTION

A core foundation of human knowledge and innovation is the ability of human experts to *research and synthesize* known facts and new findings, enabling others to comprehend, verify and build upon prior work. Recently, systems for *generative research synthesis* have emerged, promising to automate tasks that produce long-form outputs (e.g., multi-page reports), which traditionally demand hours of literature searching, reading and writing by human experts. These offerings include commercial ones—from OpenAI (OpenAI, 2025a), Gemini (Gemini, 2025a), Anthropic (Anthropic, 2025a), Grok (xAI, 2025), and Perplexity (Perplexity, 2025)—as well as open-source methods, such as STORM (Shao et al., 2024), DeepResearcher (Zheng et al., 2025), and OpenScholar (Asai et al., 2024). Existing systems demonstrate promising performance on factuality and question-answering benchmarks (Wei et al., 2024; Krishna et al., 2025; Mialon et al., 2023; Wei et al., 2025), pushing the frontier of AI capabilities.

Yet, as this new class of systems emerges, a key question remains: *how should we benchmark and evaluate generative research synthesis?* The progress of these systems requires benchmarks that carefully evaluate their critical capabilities—specifically, three core functions: (1) *retrieval*, typically from a large, complex, and constantly-evolving corpus, such as the live web, to collect key information (2) *knowledge synthesis*, to generate coherent, long-form answers that surface key facts, integrating general knowledge and findings from many retrieved sources, and (3) *verifiability*, providing citations that allow readers to trace each stated claim in the synthesized answer to a reputable source from the retrieved set. The ideal benchmark must holistically evaluate across all three of these dimensions, while providing a realistic and challenging research synthesis task.

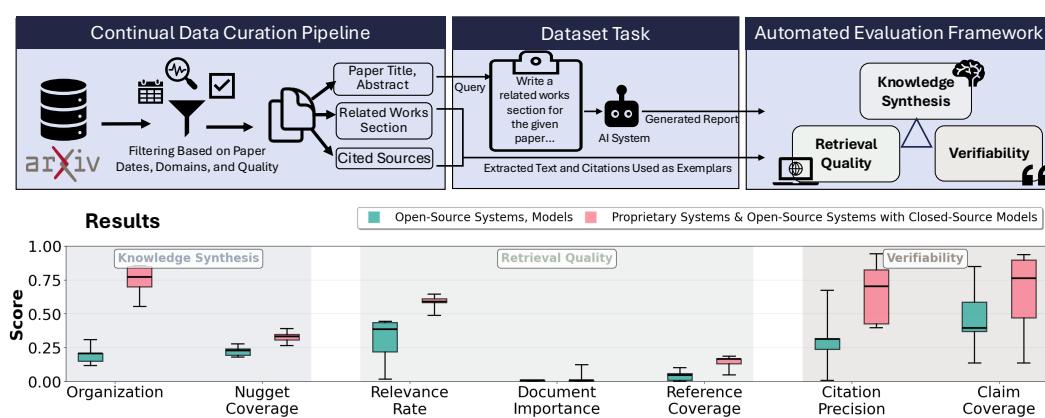


Figure 1: DeepScholarBench Overview. We propose a live, continually-updating benchmark for generative research synthesis, for which we plan to release monthly datasets and leaderboard results. We use an automated data pipeline (top left) to curate datasets from recent, high-quality ArXiv papers. Our dataset task is to generate a related works section given information about a paper (top middle). The DeepScholar-bench evaluation framework (top right) uses a holistic set of automated metrics to assess performance of system reports on three key dimensions: knowledge synthesis, retrieval quality and verifiability. We systematically evaluate 14 existing baselines (bottom) and show the performance range of them on each metric. In pink, we show the performance range of open-source systems, including DeepScholar, STORM, OpenScholar, a Search Agent and our DeepScholar-ref pipeline, each with the open-source Llama-4-Scout-17B-16E-Instruct model. In green, we show the performance range of proprietary systems and open-source systems using closed-source models, including OpenAI’s o3 DeepResearch, as well as Search Agents and DeepScholar-ref run with o3, Claude-opus-4, Gemini-2.5-pro and GPT4.1. Overall, no system surpasses a geometric mean of 31% across all metrics, reflecting significant opportunity for future work. Full evaluation results appear in Section 6.

Unfortunately, existing benchmarks fall short of these goals. Many prior works evaluate generative research synthesis systems using existing question answering benchmarks, which do not reflect realistic research synthesis tasks and instead focus on questions with short-form, easily-verifiable answers, making them severely limited for this setting (Wei et al., 2025; 2024; Mialon et al., 2023; Krishna et al., 2025; Wu et al., 2025; Wadden et al., 2020; Jin et al., 2019; Yang et al., 2018; Joshi et al., 2017; Kwiatkowski et al., 2019; Ho et al., 2020; Trivedi et al., 2022; Lee et al., 2023). These question-answering benchmarks do not capture the complexity of long-form answers synthesized from many sources, a key component of research synthesis. To address this limitation, several recent works instead leverage expert-curated datasets with open-ended research questions and exemplar answers (Asai et al., 2024; Zheng et al., 2024; you.com, 2025; Xu et al., 2025; Du et al., 2025; Su et al., 2025; Java et al., 2025). Unfortunately, these benchmarks quickly become stale and outdated as new information emerges. Furthermore, these datasets risk data contamination as new models are trained on snapshots of the web, including public datasets. The prohibitive expense of curating, maintaining, and updating expert-curated benchmarks further limits their utility towards realistic, scalable evaluation.

In this work, we introduce DeepScholar-bench, a live benchmark and holistic, automated evaluation framework designed to evaluate generative research synthesis. DeepScholar-bench draws queries from recent, high-quality ArXiv papers and focuses on a real research synthesis task: generating the related work sections of a paper by retrieving, synthesizing, and citing prior research. We plan to provide a *live* benchmark, releasing updated research queries every month, and practitioners can also run our automated data pipeline to create their own dataset instantiations. Further, we develop an automated evaluation framework that leverages human-written related works extracted from each ArXiv paper and holistically assesses performance across three key dimensions—*knowledge synthesis*, *retrieval quality*, and *verifiability*—using metrics that show strong agreement with human judgments. To promote future work, we also develop DeepScholar-ref, a simple open-source reference pipeline for generative research synthesis implemented on the LOTUS framework (lotus, 2025; Patel et al., 2025).

Using the DeepScholar-bench framework, we systematically evaluate the performance of existing systems, including open-source research synthesis systems, search agents with strong proprietary models, OpenAI DeepResearch, and DeepScholar-ref. We find that all of these existing methods exhibit significant opportunity for improvement, with no system surpassing a geometric mean of 31% across all metrics. Furthermore, on several key metrics, including Nugget Coverage, Reference Coverage and Document Importance, each evaluated method’s performance remains well below 40%, reflecting the inherent difficulty of the DeepScholar-bench task, which requires systems to navigate the live web, reasoning about the relevance and importance of documents as well

108  
109  
110  
111  
112  
113  
114  
115  
116  
117  
118  
119  
120  
121  
122  
123  
124  
125  
126  
127  
128  
129  
130  
131  
132  
133  
134  
135  
136  
137  
138  
139  
140  
141  
142  
143  
144  
145  
146  
147  
148  
149  
150  
151  
152  
153  
154  
155  
156  
157  
158  
159  
160  
161  
Table 1: Summary of Evaluation Metrics.

| Metric  | Description   |
|---|---|
| <i>Knowledge Synthesis</i>                                  |   |
| Organization & Coherency<br>Nugget Coverage                 | assesses organization and coherence of system answer<br>assesses the answer's coverage of essential facts   |
| <i>Retrieval Quality</i>                                    |   |
| Relevance Rate<br>Document Importance<br>Reference Coverage | measures avg. relevance among all referenced sources<br>measures how notable referenced sources are, using citation counts<br>assesses the referenced set's coverage of key, important references |
| <i>Verifiability</i>  |   |
| Citation Precision<br>Claim Coverage                        | measures percent of cited sources that support their accompanying claim<br>measures percent of claims that are fully supported by cited sources   |

as surfacing key facts into a cohesive final answer. Notably, OpenAI’s DeepResearch offers strong performance relative to other baselines, outperforming many prior methods on knowledge synthesis and retrieval quality, with scores of 39.2% on Nugget Coverage, 18.7% on Reference Coverage and 12.4% on Document Importance; however, it struggles to provide strong verifiability relative to many other methods. We also find that DeepScholar-ref reference pipeline represents a strong open-source baseline offering competitive performance on most metrics and up to  $6.3 \times$  higher verifiability compared to OpenAI’s DeepResearch. Nevertheless, DeepScholar-bench remains far from saturated, representing exciting opportunities for further work. We hope that our benchmark framework and reference pipeline support the progress of new systems, and we believe that resolving DeepScholar-bench represents a critical milestone towards more capable AI systems.

Overall, our main contributions are the following:

- We propose DeepScholar-bench, a live benchmark dataset with real research synthesis tasks and an automated, holistic evaluation.
- We develop DeepScholar-ref, a simple open-source reference pipeline for generative research synthesis that attains competitive performance with open-source systems, search agents, and OpenAI’s DeepResearch across many metrics using the same models.
- We perform a systematic evaluation of existing baselines on DeepScholar-bench, finding significant opportunities for improvement, with no system surpassing a geometric mean of 31% across all metrics.

## 2 RELATED WORK

**Long-form Synthesis Benchmarks.** While our work proposes a continually-updated, live benchmark using an automated data pipeline, several prior works instead provide expert-curated datasets for long-form research synthesis tasks, including ScholarQABench (Asai et al., 2024), OpenResearcher (Zheng et al., 2024), DeepConsult (you.com, 2025), ResearcherBench (Xu et al., 2025), DeepResearch Bench (Du et al., 2025), Deep Research Bench (FutureSearch et al., 2025), SurGE (Su et al., 2025), and LiveDRBench (Java et al., 2025). Unfortunately, these expert-curated benchmarks, are expensive to construct and update, can quickly become outdated, as new information becomes available, and risk data contamination, as new models are trained on publicly available data.

Alternatively, several recent benchmarks, including AcademicEval (Zhang et al., 2024b), LongBench-Cite (Zhang et al., 2024a) and SciIG (Garg et al., 2025), evaluate long-form generation tasks that *do not require search over the live web*, which is a key component of generative research synthesis and our benchmark. Other benchmarks focus on other long-form generation tasks, such as Wikipedia-like article generation (Shao et al., 2024), which differs substantially from our focus on complex research synthesis tasks. Crucially, unlike each of these prior works, our work proposes a live, continually-updated benchmark for evaluating generative research synthesis.

**Factuality and Question Answering Benchmarks.** While this work proposes a framework for studying complex, long-form research synthesis tasks, which lack an absolute notion of correctness and admit many possible reasonable answers, several recent works focus their evaluation on question answering (QA) and factuality benchmarks with short-form, easily verifiable answers. These prior benchmarks include SimpleQA (Wei et al., 2024), FRAMES (Krishna et al., 2025), GAIA (Mialon

et al., 2023), BrowserComp (Wei et al., 2025), BrowserComp-Plus (Chen et al., 2025) WebWalkerQA (Wu et al., 2025), DeepResearch Arena (Wan et al., 2025) and others traditionally used to evaluate retrieval-augmented generation (Wadden et al., 2020; Jin et al., 2019; Yang et al., 2018; Joshi et al., 2017; Kwiatkowski et al., 2019; Ho et al., 2020; Trivedi et al., 2022; Lee et al., 2023). Additionally, several benchmark develop automated dataset curation pipelines for live benchmark; however, their task focuses on short-form question-answering, as opposed to long-form report generation (Ouyang et al., 2025; Meem et al., 2024; Jiang et al., 2025).

### 3 THE DEEPSCHOLAR DATASET

We study the task of generating a related works section of an academic paper, a fundamental research synthesis task, for which we leverage human-written exemplars from our automated data pipeline to ground our evaluation. We source our dataset queries by scraping ArXiv papers (arxiv, 2025) accepted at academic conferences, and we formalize our dataset task as follows: given a paper’s description  $d$ , the goal is to retrieve a set of relevant sources  $S$  and generate a related works sections  $W$  by synthesizing and citing the retrieved documents. We briefly overview our automated data collection framework (Section 3.1) and describe the dataset instantiation (Section 3.2) used in our evaluation (Section 6). Appendix 8.1 provides additional details.

#### 3.1 AUTOMATED DATA COLLECTION FRAMEWORK

Our automated data collection framework aims to achieve the following design goals:

1. Inclusion of *diverse* paper topics across a wide variety of research domains.
2. Focus on *recent* research papers, both to provide realistic, timely benchmark queries and to prevent data contamination when benchmarking models trained on snapshots of the web.
3. Control for *quality* of the scraped ArXiv papers and extracted data, focusing on peer-reviewed manuscripts that are accepted at academic conferences.

Our data pipeline scrapes papers, filters them, and extracts content to construct datasets that include metadata about each ArXiv paper (e.g., the title, abstract, and link), the papers’ related work section, and a citation list of references found in the papers’ related work sections. Our scraper begins by loading papers from a configured set of ArXiv domains (e.g., cs.ML) and configured publication-date range. To avoid possible data contamination arising from multiple ArXiv versions, we keep only v1 ArXiv papers. To control for paper quality, our pipeline then optionally filters papers to keep only those listed as “accepted” or “published” at a conference based on the ArXiv metadata. We also exclude papers that do not have an explicit “Related Work” section and .bib file, containing well-formatted bibliography entries. Our pipeline then processes each paper to extract the Related Works section from both the LaTex files and PDF file, if available. We clean the extracted LaTex section to remove any labels and comments. We also extract all citations found in the related work section and we use the ArXiv and OpenAlex APIs (OpenAlex, 2025) to recover more detailed information, such as abstracts, authors, and links for both ArXiv and non-ArXiv references.

#### 3.2 DATASET DESCRIPTION AND STATISTICS

The dataset instantiation, DeepScholar-June-2025 used in our evaluation (Section 6) configures a publication date range between April and June 2025, following the April 5th, 2025 release date of Llama-4 models (noa, 2025), the main open-source model we benchmark. This instantiation scrapes papers from a diverse set of 18 distinct ArXiv domains—including, cs.IR, cs.CV, cs.AI, cs.CL, cs.LG, cs.DC, cs.DB, cs.AR, cs.SD, cs.CR, cs.ET, cs.GR, cs.PL, cs.SY, cs.OS, cs.PF, cs.SE, cs.MM—and selects papers marked as accepted at a conference. We additionally exclude papers with related works sections longer than 1,000 words to control for cost. Our final dataset includes 63 ArXiv papers, each providing a single query and an extracted expert-written exemplar for our evaluation. We make our scripts available to allow others to configure different datasets, and we plan to release monthly datasets of recent queries. Our experiments leverage the abstract of each paper as the paper’s description  $d$ , provided to each baseline system as context within the query. We analyze the human-written exemplars from our dataset, and we find that, on average, each related work

216 section contains 23 unique references, and we find over 63% of all cited references on ArXiv. We  
 217 provide additional experiments in Appendix 8.3.2 on a more recent, expanded dataset DeepScholar-  
 218 Nov-2025, which contains 200 queries from over 75 distinct arXiv subject areas, spanning Computer  
 219 Science, Physics, Quantitative Biology, Economics, and Quantitative Finance. Our results on this  
 220 dataset confirm the generalization of our benchmark and main experimental conclusions.  
 221

## 222 4 THE DEEPSCHOLAR EVALUATION FRAMEWORK 223

224 Evaluating research synthesis is inherently challenging due to the task’s complexity, and the lack  
 225 of a simple, “ground truth” notion of correctness, permitting many possible answers. To address  
 226 these challenges, we propose a holistic, automated evaluation that assesses 7 fine-grained metrics  
 227 across three key dimensions that reflect the core capabilities of research synthesis: *knowledge syn-*  
 228 *thesis* (Section 4.1), *retrieval quality* (Section 4.2), and *verifiability* (Section 4.3). We overview  
 229 our evaluation framework in this section, and provide further details and analysis of our metrics in  
 230 Appendix 8.4, as well as manual validation of our LLM-based metrics in Appendix 8.3.  
 231

### 232 4.1 KNOWLEDGE SYNTHESIS 233

234 Knowledge synthesis reflects the ability of a system to generate an effective final report which sur-  
 235 faces key facts and information into a coherent writeup. We evaluate both the overall *Organization*  
 236 and *Coherency* of each system report, as well as its factual content according to its *Nugget Coverage*.  
 237

238 **Organization and Coherency.** We use an LLM-as-a-judge to assess organization and coherence and  
 239 perform pairwise comparisons between the system generated report and the human-written exemplar  
 240 from the dataset. We permute each evaluated pair of reports to avoid position bias (Li et al., 2025),  
 241 and we report the win-rate of each baseline. This model-based evaluation provides scalability while  
 242 also serving as a strong surrogate for human preferences (Rahmani et al., 2024b; Li et al., 2024;  
 243 2025; Arabzadeh and Clarke, 2025), which we validate in our experiments in Appendix 8.3.  
 244

245 **Nugget Coverage.** To assess the quality of the information content of a generated report, we use  
 246 a nugget-based evaluation. An *information nugget* is an essential fact or component relevant for an  
 247 answer (Pradeep et al., 2025; Upadhyay et al., 2024b;a; Faggioli et al., 2023; Rahmani et al., 2024a).  
 248 For our task, we generate nuggets from the human-written exemplar related-work section for each  
 249 query, and for each generated report, we compute the nugget coverage score, the fraction of nuggets  
 250 present in the answer, following the automated, LLM-based methodology of Pradeep et al. (2025).  
 251

### 252 4.2 RETRIEVAL QUALITY 253

254 A key component of generative research synthesis is retrieval over the live web, which differs sub-  
 255 stantially from traditional information retrieval evaluations (Thakur et al., 2021; Nanni et al., 2017).  
 256 This setting lacks a closed corpus with gold labels — expert-written exemplars provide *one* reason-  
 257 able reference set, but there may be many possible alternative sets that are likewise high-quality. To  
 258 address these challenges, we evaluate three metrics of each generated report’s retrieved reference  
 259 set: the relevance rate, reference coverage of key sources, and document importance.  
 260

261 **Relevance Rate.** We asses the relevance of each retrieved document, following the Cranfield  
 262 model (Voorhees, 2009), which is standard in IR evaluations and considers relevance of individual  
 263 documents given a query, independent of other documents. We use an LLM-as-a-judge approach to  
 264 assign graded relevance scores to each retrieved source, following recent works (Upadhyay et al.,  
 265 2024b; Faggioli et al., 2023; Rahmani et al., 2024b; Thomas et al., 2024; Asai et al., 2024). Specifi-  
 266 cally, the LLM-judge assigns a relevance score,  $Rel(s)$ , from 0–2 for each source,  $s$ , in the retrieved  
 267 set  $S$ , and we compute the relevance rate of  $S$  as:  
 268

$$269 RR(S) = \frac{1}{2|S|} \sum_{s \in S} Rel(s).$$

270 **Reference Coverage.** We introduce a metric to measure the *reference coverage* of each report’s  
 271 retrieved set. A key challenge in measuring this value is in defining a core set of “important” sources  
 272 that a good report should reference. To build this set, we take all references from the high-quality,  
 273 human-written exemplar and label each as either “important” or “not-important”, considering a “not-  
 274 important” reference as one that could be omitted or substituted by a different source. For each  
 275

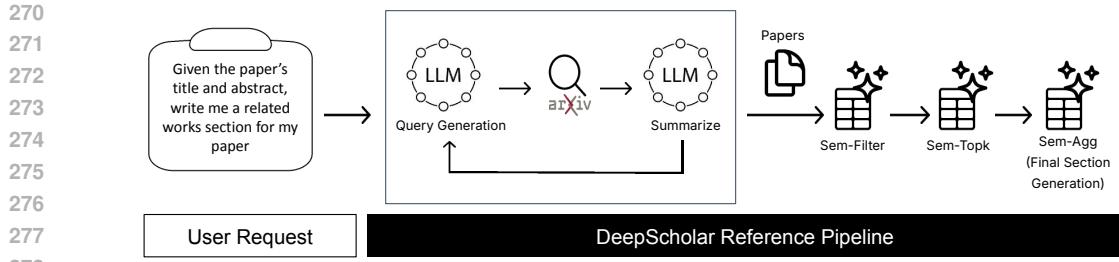


Figure 2: Overview of DeepScholar-ref. The system iteratively writes queries and performs web search, before passing the search results through series of semantic operators using the LOTUS system for LLM-based data-processing, including filtering step to discard irrelevant sources, a top-k ranking step to find most relevant sources, and an aggregation step to generate the final report from all remaining sources.

generated report, we then report its reference coverage by dividing the number of retrieved important references by total number of important references. We follow the below formula, where  $E$  is the set of "important" references from the human-written exemplar:

$$RC(S, E) = \frac{1}{|E|} \sum_{s \in S} I[s \in E].$$

**Document Importance.** While the above retrieval metrics assess topical relevance and coverage of key references, an ideal research synthesis system must also retrieve many *notable and important* sources. Exemplar human-written reports typically contain ample references of primary-sources and highly-cited academic publications. We compute the *document importance* of a retrieved set by considering the number of citations of each of its sources. Specifically, we consider the median number of citations over sources in  $S$ , compared to this the median number of citations over sources in the reference set,  $S^*$ , from the human-written exemplar, and set an upper-bound of 1, given by:

$$DI(S, S^*) = \min \left( \frac{\text{median}\{\text{num-cites}(s) | s \in S\}}{\text{median}\{\text{num-cites}(s^*) | s^* \in S^*\}}, 1 \right),$$

where  $\text{num-cites}(s)$  is the number of citations for source  $s$ .

#### 4.3 VERIFIABILITY

To evaluate the verifiability generated reports using the citation precision and claim coverage with LLM-based entailment evaluations, following prior work (Gao et al., 2023; Liu et al., 2023).

**Citation Precision.** We measure sentence-level precision, where a citation is considered precise if the referenced source supports at least one claim made in the accompanying sentence. For a full report, we compute citation precision by averaging the precision of each citation.

**Claim Coverage.** Claim coverage of a report is computed by assigning a sentence-level score—of one if the sentence’s cited sources support all claims made in the sentence—and averaging all sentence-level scores in the report. We make two adaptations to the original definition of prior work (Gao et al., 2023; Worledge et al., 2024; Liu et al., 2023) to tailor our metric to our long-form synthesis task. First, we relax the original claim coverage definition to consider a sliding window of sentences with supporting references. Specifically, we assign a sentence-level coverage score of 1 if the sentence is fully supported by the sources cited either within the sentence or in a window of  $w$  preceding or following sentences. Additionally, since our task query provides context describing a paper, we consider this context as an implicitly cited reference for each sentence.

## 5 DEEPSCHOLAR-REF

We introduce DeepScholar-ref, an open-source reference pipeline designed for generative research synthesis. As Figure 2 shows, DeepScholar-ref takes a user’s query and iteratively generates web-search queries, summarizing search results in each round before generating new queries. The system then post-processes the search results leveraging a series of semantic operators (Patel et al., 2025) implemented on the LOTUS API (lotus, 2025). This includes a semantic filtering step, which leverages an LLM to filter out irrelevant source documents, followed by a semantic top-k which performs an LLM-based ranking over the documents based on their relevance to the user query. Finally, we perform a semantic aggregation over the final source documents to generate the final report. We provide further details of each step of our reference pipeline in Appendix Section 8.6.

324 Table 2: Main Results. **The best baseline** is shown in bold and the second-best baseline is underlined. \* indicates that the  
 325 **best baseline** is statistically significantly better than the second-best baseline under a paired two-tailed t-test with  $p < 0.05$ .  
 326

|                                       | Knowledge Synthesis |               | Retrieval Quality |               |               | Verifiability     |                       | Geo. Mean         |
|---------------------------------------|---------------------|---------------|-------------------|---------------|---------------|-------------------|-----------------------|-------------------|
|                                       | Org.                | Nug. Cov.     | Rel. Rate         | Ref Cov.      | Doc Imp.      | Cite-P            | Claim Cov ( $w = 1$ ) |                   |
| <i>Human-written Exemplars</i>        |                     |               |                   |               |               |                   |                       |                   |
| Human-written Exemplars               | .500                | 1.000         | .585              | 1.000         | 1.000         | .900 <sup>1</sup> | .850 <sup>1</sup>     | .782 <sup>1</sup> |
| <i>Open Source Research Systems</i>   |                     |               |                   |               |               |                   |                       |                   |
| DeepResearcher (Llama-4)              | .206                | .230          | .385              | .047          | .008          | .312              | .396                  | .137              |
| STORM (Llama-4)                       | .119                | .183          | .218              | .003          | .006          | .238              | .586                  | .073              |
| OpenScholar (Llama-4)                 | .309                | .278          | .017              | .008          | .013          | .010              | .138                  | .042              |
| <i>Search Agents</i>                  |                     |               |                   |               |               |                   |                       |                   |
| Search Agent (Llama-4)                | .151                | .193          | .445              | .060          | .009          | .316              | .368                  | .135              |
| Search Agent (GPT-4.1)                | .556                | .265          | .490              | .050          | .009          | .498              | .470                  | .186              |
| Search Agent (o3)                     | .849                | .348          | .610              | .165          | .026          | .425              | .495                  | .287              |
| Search Agent (Claude)                 | .698                | .307          | .583              | .131          | .008          | .701              | .760                  | .256              |
| Search Agent (Gemini)                 | .706                | .277          | .583              | .061          | .010          | .415              | .398                  | .196              |
| <i>Commercial Systems</i>             |                     |               |                   |               |               |                   |                       |                   |
| OpenAI DeepResearch                   | <b>.857</b>         | <u>.392</u> * | <u>.629</u>       | <b>.187</b> * | <b>.124</b> * | .399              | .138                  | <b>.309</b> *     |
| <i>DeepScholar Reference Pipeline</i> |                     |               |                   |               |               |                   |                       |                   |
| DeepScholar-ref (Llama-4)             | .206                | .241          | .436              | .103          | .008          | .674              | .851                  | .195              |
| DeepScholar-ref (GPT-4.1)             | .809                | .348          | .590              | .166          | .008          | .788              | .899                  | .285              |
| DeepScholar-ref (GPT-4.1, o3)         | <b>.857</b>         | <u>.384</u>   | <b>.645</b>       | .167          | .007          | .824              | .760                  | .285              |
| DeepScholar-ref (GPT-4.1, Claude)     | .698                | .307          | .610              | .152          | .009          | <b>.944</b> *     | .895                  | .286              |
| DeepScholar-ref (GPT-4.1, Gemini)     | .770                | .331          | .590              | <u>.181</u>   | .006          | <b>.904</b>       | <b>.937</b> *         | .282              |

<sup>1</sup> The automated verifiability metrics in our evaluation under-estimate the actual verifiability of human writing, thus, we provide an estimate using manual validation over a small sample, and we exclude them from the geometric mean for the human-written exemplars. This is because Citation Precision and Claim Coverage require us to assess entailment relations between claims and cited reference. For each LLM-based system, we are able to track the precise snippet and context from cited sources, which are directly fed as context to the LLM. On the other hand, for the human-written exemplars, we lack gold labels pointing to the precise snippet of text that each reference refers to. Our measurements for the human-written exemplars instead rely on the title and abstract of each cited source as a proxy.

## 350 6 EXPERIMENTAL RESULTS

352 In this section, we evaluate recent state-of-the-art generative research systems as well as  
 353 DeepScholar-ref on DeepScholar-bench. Overall, we find the following:

354

- 355 Existing baselines for generative research synthesis, including strong open-source LLM  
 356 systems, search agents, and commercial systems, demonstrate significant room for im-  
 357 provement across all three key dimensions: knowledge synthesis, retrieval quality and ver-  
 358ifiability. Specifically, no system surpasses a geometric mean of 31% across all metrics.
- 359 DeepScholar-ref provides a strong baseline, consistently improving upon the performance  
 360 of prior open-source systems and search agents, as well as achieving competitive per-  
 361 formance and up to 6.3× higher verifiability compared to OpenAI’s DeepResearch.

362 **Experimental Setup.** We benchmark open-source research systems, including DeepRe-  
 363 searcher (Zheng et al., 2025), STORM (Shao et al., 2024) and OpenScholar (Asai et al., 2024),  
 364 search agents, with Llama-4-Scout-17B-16E-Instruct (Meta, 2025), GPT-4.1-2025-04-14 (Ope-  
 365 nAI, 2025b), o3-2025-04-16 (OpenAI, 2025b), Claude-opus-4-20250514 (Anthropic, 2025b),  
 366 and Gemini-2.5-pro (Gemini, 2025b) models, OpenAI’s o3-deep-research (OpenAI, 2025b), and  
 367 DeepScholar-ref. For each benchmarked method, we control the retrieval corpus by allowing each  
 368 system to access the Web only through the ArXiv API (arxiv, 2025). We additionally avoid possible  
 369 information leakage during search by filtering out any search results that were published after the  
 370 query paper’s publication date. We provide further details of our setup in Appendix 8.2.

### 372 6.1 MAIN RESULTS

374 Table 2 provides of each method’s performance on DeepScholar-June-2025. For each baseline, we  
 375 report metrics averaged over all queries, and the geometric mean of all metrics. We also provide  
 376 additional results in Appendix 8.3—including metadata statistics of the generated reports (Table 6),  
 377 statistics related to our evaluation metrics (Table 7), and manual validation of our LLM-based met-  
 378 rics (Table 10)—and example reports in Appendix 8.7. We discuss key findings in detail below.

378  
379  
3806.1.1 GENERATIVE RESEARCH SYNTHESIS SYSTEMS DEMONSTRATE LARGE ROOM FOR  
IMPROVEMENT.381  
382  
383  
384  
385  
386  
387

From Table 2, we see that no system surpasses a geometric mean of .31 across all metrics, with OpenAI DeepResearch obtaining the highest geometric mean. Moreover, on several key metrics, including Nugget Coverage, Reference Coverage and Document Importance, each baseline’s performance remains below .40. This reflects the inherent difficulty of the generative research task provided by DeepScholar-bench, particularly the need for systems to navigate the live web, reasoning about coverage and importance of documents, before surfacing key information in long-form report.

388  
389  
390  
391  
392  
393  
394  
395  
396

We now analyze each evaluated dimension, comparing performance of the open-source research systems, search agents and commercial systems to the human-written exemplars. On Knowledge Synthesis, we see that OpenAI DeepResearch offers the best performance compared to all other prior methods on both Organization, with a score of .857, and Nugget Coverage, with a score of .392. OpenAI DeepResearch, as well as the o3, Claude and Gemini search agents achieve relatively high Organization scores compared to human-written exemplars. However, on Nugget Coverage all prior methods scores below .40. This demonstrates that while existing systems, especially those using state-of-the-art models, can generate well-organized and coherent summaries, they still struggle to surface key facts to answer the research query, a crucial capability for research synthesis tasks.

397  
398  
399  
400  
401  
402  
403  
404

Turning our attention to the Retrieval Quality performance of prior methods, we once again find significant room for improvement. Once again, OpenAI DeepResearch offers the strongest performance among the other benchmarked prior methods on Relevance Rate, Reference Coverage and Document Importance, but still far from saturates performance. While it’s Relevance Rate shows strong performance, exceeding that of the human exemplars with a score of .629, it’s Reference Coverage and Document Importance scores remain exceedingly low: .187 and .124 respectively. This demonstrates that while state-of-the-art generative research synthesis systems excel at retrieving *relevant* sources, they still struggle to find *comprehensive sets of notable sources*, falling short compared to the ability of human experts.

405  
406  
407  
408  
409  
410  
411  
412  
413

Lastly, we analyze the Verifiability performance of prior methods. We see that OpenAI DeepResearch is outperformed on both Citation Precision and Claim Coverage by the search agents with GPT4.1, o3, Claude and Gemini models. The Claude search agent offers the highest Citation Precision, a score of .701 and Claim Coverage, a score of .760. Meanwhile, OpenAI’s DeepResearch as well as the all other prior methods are unable to achieve a Citation Precision score beyond .50 and a Claim Coverage score beyond .60. We also note that the human-written exemplars appear to exhibit rather low Citation Precision and Claim Coverage scores, however these scores are under-estimate the actual verifiability of human writing<sup>1</sup>. Overall, we see that prior LLM-based systems exhibit significant headroom for improvement.

414  
415  
4166.1.2 DEEPSCHOLAR-REF PROVIDES A STRONG BASELINE FOR GENERATIVE RESEARCH  
SYNTHESIS.417  
418  
419  
420  
421  
422  
423  
424  
425

We compare the performance of DeepScholar-ref to OpenAI DeepResearch, search agents and open-source systems, finding that DeepScholar-ref provides a strong baseline with competitive performance against the other baselines across most metrics using the same or cheaper models. In comparison to OpenAI DeepResearch, DeepScholar-ref (GPT-4.1, o3) achieves a similar or higher Organization, Nugget Coverage, Relevance Rate, Reference Coverage, Citation Precision and Claim Coverage score. Notably, DeepScholar-ref achieves up to 6.3 $\times$  higher Verifiability scores but its Document Importance remain relatively low compared to OpenAI’s DeepResearch. In Appendix Table 6, we provide additional cost analysis, finding that DeepScholar-ref (GPT-4.1, o3) offers an efficient reference pipeline that is 4.3 $\times$  cheaper and 2.28 $\times$  faster than OpenAI DeepResearch.

426  
427  
428  
429  
430  
431

Next, we compare the performance of DeepScholar-ref and search agents, finding DeepScholar-ref offers competitive and often stronger performance—specifically, averaged across the 5 baselines, using the same primary model, DeepScholar-ref increases Organization by 1.18 $\times$ , Nugget Coverage by 1.17 $\times$ , the Relevance Rate by 1.06 $\times$ , Reference Coverage by 2.03 $\times$ , Citation Precision by 1.83 $\times$  and Claim Coverage by 1.86 $\times$ . Lastly, we compare DeepScholar-ref (Llama-4) to the open-source research systems, all run with the Llama-4. We see that the prior open-source research systems exhibit trade-offs among the Knowledge Synthesis, Retrieval Quality and Verifiability dimensions.

432 Table 3: Ablation Study Comparing The Effect of Different Retrieval APIs. **The best baseline** is shown in  
 433 **bold** and the second-best baseline is underlined. \* indicates that the best baseline is statistically significantly  
 434 better than the second-best baseline under a paired two-tailed t-test with  $p < 0.05$ .

|  | Knowledge Synthesis |              | Retrieval Quality |               |              | Verifiability | Geo. Mean             |             |
|--|---------------------|--------------|-------------------|---------------|--------------|---------------|-----------------------|-------------|
|  | Org                 | Nug. Cov.    | Rel. Rate         | Ref Cov.      | Doc Imp.     | Cite-P        | Claim Cov ( $w = 1$ ) |             |
| <i>DeepScholar-ref (GPT-4.1, Claude)</i> |                     |              |                   |               |              |               |                       |             |
| arxiv.org Retrieval                      | .698                | .307         | .610              | .152          | .009         | <b>.944</b>   | <u>.895</u>           | .286        |
| parallel.ai Retrieval                    | <u>.865</u>         | .444         | <u>.675</u>       | <u>.160</u>   | .017         | .846          | .781                  | .334        |
| tavily.com Retrieval                     | <b>.929*</b>        | .327         | .550              | .070          | .015         | .711          | .578                  | .258        |
| Oracle Retrieval (arxiv.org)             | .782                | <u>.487</u>  | <b>.686</b>       | <b>1.000*</b> | <b>1.000</b> | <b>.955</b>   | <b>.899*</b>          | <b>.808</b> |
| Oracle Retrieval (All)                   | .778                | <u>.528*</u> | <u>.680</u>       | <b>1.000*</b> | <u>.822</u>  | .941          | .828                  | <u>.782</u> |
| <i>DeepScholar-ref (Llama-4)</i>         |                     |              |                   |               |              |               |                       |             |
| arxiv.org Retrieval                      | <u>.206</u>         | .241         | .436              | .103          | .008         | <u>.674</u>   | .851                  | .195        |
| parallel.ai Retrieval                    | <b>.246</b>         | .265         | .559              | <u>.114</u>   | .015         | .223          | .543                  | .186        |
| tavily.com Retrieval                     | .111                | .229         | .532              | .030          | .016         | .442          | .676                  | .153        |
| Oracle Retrieval (arxiv.org)             | .202                | <u>.316</u>  | <u>.681</u>       | <b>1.000*</b> | <b>1.000</b> | .658          | <u>.868</u>           | <u>.590</u> |
| Oracle Retrieval (All)                   | .198                | <u>.350</u>  | <b>.693</b>       | <b>1.000*</b> | <u>.822</u>  | <b>.796*</b>  | <b>.890</b>           | <u>.600</u> |

441 Compared to the best-performing prior open-source methods for each metric, DeepScholar-ref offers competitive Knowledge Synthesis performance,  $1.09\times$  higher Relevance Rates,  $2.18\times$  higher Reference Coverage,  $2.08\times$  higher Citation Precision and  $1.41\times$  higher Claim Coverage.

442 Overall, the strong *relative* performance of DeepScholar-ref likely reflects the efficiency of the data-  
 443 processing semantic operators (Patel et al., 2025) that DeepScholar-ref uses to perform LLM-based  
 444 filtering, ranking and summarization of sources to generate its report. Notably, DeepScholar-ref  
 445 still demonstrates significant room for improvement and far from saturates DeepScholar-Bench,  
 446 especially on key Knowledge Synthesis and Retrieval Quality metrics.

## 447 6.2 UNDERSTANDING OPPORTUNITIES FOR IMPROVEMENT.

448 To analyze performance and opportunities for improvement, we conduct an ablation study, testing  
 449 different retrievers, including two oracle settings. Table 3 shows the performance of DeepScholar-ref  
 450 (GPT-4.1, Claude) and DeepScholar-ref (Llama-4), each with 3 different retrieval APIs: arxiv.org,  
 451 the default in our main results, parallel.ai and tavily.com. In addition, our two oracle retrievers  
 452 include the Oracle Retrieval (arxiv.org) setting and the Oracle Retrieval (All) setting, which pro-  
 453 vide the system with *ArXiv* references and *all* references, respectively, from the set of important  
 454 references cited in exemplars, following our methodology for evaluating Reference Coverage.

455 Overall, the results demonstrate that key opportunities for improvement lie in both retrieval and  
 456 knowledge synthesis capabilities. First, we see that DeepScholar-ref (GPT-4.1, Claude) with ei-  
 457 ther oracle retriever nearly saturates performance on Retrieval Quality and Verifiability metrics,  
 458 whereas the same method, using the arxiv.org, parallel.ai or tavily.com retrievers, score much lower.  
 459 Specifically, significant performance gaps exist for Reference Coverage and Document Importance,  
 460 demonstrating the system struggles to navigate the live web and recover a diverse set of key,  
 461 notable sources. Additionally, we also see that oracle retrievers improve the Nugget Coverage of either  
 462 DeepScholar-ref methods by up to  $1.62\times$  compared to the arxiv.org, parallel.ai or tavily.ai retriev-  
 463 ers. Yet, the oracle settings still far from saturate Nugget Coverage, highlighting the AI system still  
 464 struggle to effectively surface important facts and insights, even with high-quality sources.

## 465 6.3 HUMAN EVALUATION

466 To assess whether LLM-based metrics reflect human expert judgments, we conduct a human eval-  
 467 uation with 11 annotators, all Computer Science PhD students from four research universities across  
 468 North America. In total, we collect over 300 human annotations in order to validate the LLM-based  
 469 judges introduced by our automated evaluation for assessing knowledge synthesis and retrieval qual-  
 470 ity. We provide further details of our setup in Appendix 8.3.4.

471 **Agreement analysis.** Table 4 shows the results of human evaluation study as confusion matrices  
 472 taken between the majority vote of human annotators and the LLM-judge. Overall, the results show  
 473 the robustness of our LLM-judges based on their strong alignment with the expert human annota-  
 474 tors. Specifically, we observe a 71.43% agreement score for pairwise comparisons judging Organi-  
 475 zation, a 83.33% agreement score for nugget labeling to compute Nugget Coverage, and a 65.9%

486  
 487  
 488  
 489  
 490 Table 4: Confusion matrices comparing human and LLM judgments on organization, Nugget im-  
 491 portance judgments, and Reference-importance judgments, where in each table rows and columns  
 492 represent human and LLM judgments.  
 493  
 494

| Organization |           |           | Nugget Importance |             |        | Reference Importance |             |          |       |
|--------------|-----------|-----------|-------------------|-------------|--------|----------------------|-------------|----------|-------|
| Human / LLM  | Reference | Generated | Tie               | Human / LLM | Vital  | Okay                 | Human / LLM | Not Imp. | Imp.  |
| Reference    | 14.29%    | 0%        | 14.29%            | Vital       | 58.33% | 8.33%                | Not Imp.    | 40.2%    | 9.8%  |
| Generated    | 0%        | 57.14%    | 14.29%            | Okay        | 8.33%  | 25.00%               | Imp.        | 24.2%    | 25.7% |
| Tie          | 0%        | 0%        | 0%                | Irrelevant  | 0%     | 0%                   |             |          |       |

495  
 496 agreement score for labeling reference importance to compute Reference Coverage. Notably, each  
 497 of these tasks require reasoning about complex academic literature and lengthy candidate related  
 498 works sections. The observed agreement rates from our study provide promising results, validating  
 499 the use of automated LLM-judges and metrics to assess complex generative research synthesis tasks.  
 500

501 For Organization, we see from the confusion matrix that the main point of human- and LLM-judges  
 502 most often agree on pairwise comparisons, with strong disagreements (i.e., humans preferring the  
 503 reference report and the LLM preferring the generated report or vice versa) are rare. Moreover, of the  
 504 disagreements that occur between human- and LLM-judges, the LLM mis-judgments are relatively  
 505 equally balanced between picking the Reference report and the Generated Report.  
 506

507 For Nugget Importance, we observe that in addition to the strong agreement rate observed from the  
 508 confusion matrix, we also see that the human majority vote find all LLM-generated nuggets to be  
 509 relevant, indicating hallucinations are rare. Moreover we see that the false positive and false negative  
 510 rate of the LLM-judge are similar, and both rather small, less than 10%, once again indicating that  
 511 severe LLM mis-labeling is rather rare.  
 512

513 Finally, for Reference Importance we observe an overall agreement score of 65.9%, and importantly  
 514 false negative rate of 9.8%, i.e., when the LLM incorrectly labels a reference as important. The  
 515 low false negative rate indicates a low likelihood of our Reference Coverage metric falsely penal-  
 516 izing systems. The larger off-diagonal mass (24.2%) reflects under-labeling of essential references  
 517 by the LLM, suggesting that our Reference Coverage scores provide a rather conservative metric,  
 518 measuring "recall" of only a subset of all truly important references for each query.  
 519

## 520 7 CONCLUSION

521 In this work, we introduced DeepScholar-bench, a live dataset and holistic, automated evaluation  
 522 framework designed to rigorously benchmark an emerging class of systems designed for generative  
 523 research synthesis. By automatically sourcing queries from high-quality, recent ArXiv papers, our  
 524 benchmark mitigates the risks of data staleness and training contamination, while offering a real  
 525 research synthesis task. Moreover, DeepScholar-bench provides an automated evaluation to holis-  
 526 tically measure three critical dimensions: retrieval quality, knowledge synthesis and verifiability.  
 527 We further release DeepScholar-ref, a reference pipeline, which we find provides a strong baseline  
 528 for generative research synthesis. Overall our systematic evaluation of prior open-source systems,  
 529 search agents, OpenAI’s DeepResearch and DeepScholar-ref demonstrates significant opportunities  
 530 for future work, with no system surpassing a geometric mean of 31% across all metrics. These re-  
 531 sults demonstrate both the difficulty of DeepScholar-bench and the exciting opportunity for further  
 532 advancement in this space. We hope that DeepScholar-bench and DeepScholar-ref will support the  
 533 development of more capable AI systems for generative research synthesis.  
 534

## 535 REFERENCES

536 2025. Llama 4 - a meta-llama Collection. <https://huggingface.co/collections/meta-llama/llama-4-67f0c30d9fe03840bc9d0164>  
 537 Salaheddin Alzubi, Creston Brooks, Purva Chiniya, Edoardo Contente, Chiara von Gerlach, Lucas  
 538 Irwin, Yihan Jiang, Arda Kaz, Windsor Nguyen, Sewoong Oh, Himanshu Tyagi, and Pramod  
 539 Viswanath. 2025. Open Deep Search: Democratizing Search with Open-source Reasoning  
 540 Agents. <https://arxiv.org/abs/2503.20201v1>

540 Anthropic. 2025a. Claude takes research to new places. <https://www.anthropic.com/news/research>  
 541  
 542 Anthropic. 2025b. Introducing Claude 4. <https://www.anthropic.com/news/clause-4>  
 543  
 544 Negar Arabzadeh and Charles L. A. Clarke. 2025. Benchmarking LLM-based Re-  
 545 levance Judgment Methods. In *Proceedings of the 48th International ACM SIGIR*  
 546 Conference on Research and Development in Information Retrieval (Padua,  
 547 Italy) (SIGIR '25). Association for Computing Machinery, New York, NY, USA.  
 548 <https://doi.org/10.1145/3726302.3730305>doi:10.1145/3726302.3730305  
 549  
 550 arxiv. 2025. arXiv.org e-Print archive. <https://arxiv.org/>  
 551  
 552 Akari Asai, Jacqueline He, Rulin Shao, Weijia Shi, Amanpreet Singh, Joseph Chee Chang, Kyle  
 553 Lo, Luca Soldaini, Sergey Feldman, Mike D'arcy, David Wadden, Matt Latzke, Minyang  
 554 Tian, Pan Ji, Shengyan Liu, Hao Tong, Bohao Wu, Yanyu Xiong, Luke Zettlemoyer, Gra-  
 555 ham Neubig, Dan Weld, Doug Downey, Wen-tau Yih, Pang Wei Koh, and Hannaneh Ha-  
 556 jishirzi. 2024. OpenScholar: Synthesizing Scientific Literature with Retrieval-augmented LMs.  
 557 <https://doi.org/10.48550/arXiv.2411.14199>doi:10.48550/arXiv.2411.14199 arXiv:2411.14199  
 [cs].  
 558  
 559 Zijian Chen, Xueguang Ma, Shengyao Zhuang, Ping Nie, Kai Zou, Andrew Liu, Joshua Green,  
 560 Kshama Patel, Ruoxi Meng, Mingyi Su, Sahel Sharifmoghaddam, Yanxi Li, Haoran Hong,  
 561 Xinyu Shi, Xuye Liu, Nandan Thakur, Crystina Zhang, Luyu Gao, Wenhui Chen, and Jimmy  
 562 Lin. 2025. BrowseComp-Plus: A More Fair and Transparent Evaluation Benchmark of Deep-  
 563 Research Agent. <https://doi.org/10.48550/arXiv.2508.06600>doi:10.48550/arXiv.2508.06600  
 564 arXiv:2508.06600 [cs].  
 565  
 566 Mingxuan Du, Benfeng Xu, Chiwei Zhu, Xiaorui Wang, and Zhendong Mao. 2025.  
 567 DeepResearch Bench: A Comprehensive Benchmark for Deep Research Agents.  
 568 <https://doi.org/10.48550/arXiv.2506.11763>doi:10.48550/arXiv.2506.11763 arXiv:2506.11763  
 [cs].  
 569  
 570 Guglielmo Faggioli, Laura Dietz, Charles Clarke, Gianluca Demartini, Matthias Hagen, Clau-  
 571 dia Hauff, Noriko Kando, Evangelos Kanoulas, Martin Potthast, Benno Stein, and Henning  
 572 Wachsmuth. 2023. Perspectives on Large Language Models for Relevance Judgment. In *Proceed-  
 573 ings of the 2023 ACM SIGIR International Conference on Theory of Information Retrieval*. 39–50.  
 574 <https://doi.org/10.1145/3578337.3605136>doi:10.1145/3578337.3605136 arXiv:2304.09161 [cs].  
 575  
 576 FutureSearch, Nikos I. Bosse, Jon Evans, Robert G. Gambee, Daniel Hnyk,  
 577 Peter Mühlbacher, Lawrence Phillips, Dan Schwarz, and Jack Wildman.  
 578 2025. Deep Research Bench: Evaluating AI Web Research Agents.  
 579 <https://doi.org/10.48550/arXiv.2506.06287>doi:10.48550/arXiv.2506.06287 arXiv:2506.06287  
 [cs].  
 580  
 581 Tianyu Gao, Howard Yen, Jiatong Yu, and Danqi Chen. 2023. Enabling Large Language Models  
 582 to Generate Text with Citations. In *Proceedings of the 2023 Conference on Empirical Methods in  
 583 Natural Language Processing*, Houda Bouamor, Juan Pino, and Kalika Bali (Eds.). Association  
 584 for Computational Linguistics, Singapore, 6465–6488. [https://doi.org/10.18653/v1/2023.emnlp-main.398](https://doi.org/10.18653/v1/2023.emnlp-<br/>
  585 main.398)doi:10.18653/v1/2023.emnlp-main.398  
 586  
 587 Krishna Garg, Firoz Shaik, Sambaran Bandyopadhyay, and Cornelia Caragea. 2025. Let's  
 588 Use ChatGPT To Write Our Paper! Benchmarking LLMs To Write the Introduction of  
 589 a Research Paper. <https://doi.org/10.48550/arXiv.2508.14273>doi:10.48550/arXiv.2508.14273  
 590 arXiv:2508.14273 [cs].  
 591  
 592 Gemini. 2025a. Gemini Deep Research — your personal research assistant.  
 593 <https://gemini.google/overview/deep-research/>  
 594  
 595 Gemini. 2025b. Gemini models | Gemini API. <https://ai.google.dev/gemini-api/docs/models>  
 596  
 597 Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. 2020. Construct-  
 598 ing A Multi-hop QA Dataset for Comprehensive Evaluation of Reasoning Steps. In *Pro-  
 599 ceedings of the 28th International Conference on Computational Linguistics*, Donia Scott,

594 Nuria Bel, and Chengqing Zong (Eds.). International Committee on Computational Lin-  
 595 guistics, Barcelona, Spain (Online), 6609–6625. <https://doi.org/10.18653/v1/2020.coling-main.580>  
 596

597 Abhinav Java, Ashmit Khandelwal, Sukruta Midgeshi, Aaron Halfaker, Amit Deshpande,  
 598 Navin Goyal, Ankur Gupta, Nagarajan Natarajan, and Amit Sharma.  
 599 2025. Characterizing Deep Research: A Benchmark and Formal Definition.  
 600 <https://doi.org/10.48550/arXiv.2508.04183> doi:10.48550/arXiv.2508.04183 arXiv:2508.04183  
 601 [cs] version: 1.  
 602

603 Mohan Jiang, Jin Gao, Jiahao Zhan, and Dequan Wang. 2025. MAC: A Live  
 604 Benchmark for Multimodal Large Language Models in Scientific Understanding.  
 605 <https://doi.org/10.48550/arXiv.2508.15802> doi:10.48550/arXiv.2508.15802 arXiv:2508.15802  
 606 [cs].

607 Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William Cohen, and Xinghua Lu. 2019. PubMedQA: A  
 608 Dataset for Biomedical Research Question Answering. In *Proceedings of the 2019 Conference on*  
 609 *Empirical Methods in Natural Language Processing and the 9th International Joint Conference*  
 610 *on Natural Language Processing (EMNLP-IJCNLP)*, Kentaro Inui, Jing Jiang, Vincent Ng, and  
 611 Xiaojun Wan (Eds.). Association for Computational Linguistics, Hong Kong, China, 2567–2577.  
 612 <https://doi.org/10.18653/v1/D19-1259> doi:10.18653/v1/D19-1259

613 Mandar Joshi, Eunsol Choi, Daniel Weld, and Luke Zettlemoyer. 2017. TriviaQA: A Large Scale  
 614 Distantly Supervised Challenge Dataset for Reading Comprehension. In *Proceedings of the 55th*  
 615 *Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*,  
 616 Regina Barzilay and Min-Yen Kan (Eds.). Association for Computational Linguistics, Vancouver,  
 617 Canada, 1601–1611. <https://doi.org/10.18653/v1/P17-1147> doi:10.18653/v1/P17-1147

618 Satyapriya Krishna, Kalpesh Krishna, Anhad Mohananey, Steven Schwarcz,  
 619 Adam Stambler, Shyam Upadhyay, and Manaal Faruqui. 2025. Fact, Fetch,  
 620 and Reason: A Unified Evaluation of Retrieval-Augmented Generation.  
 621 <https://doi.org/10.48550/arXiv.2409.12941> doi:10.48550/arXiv.2409.12941 arXiv:2409.12941  
 622 [cs].

623 Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh,  
 624 Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina  
 625 Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit,  
 626 Quoc Le, and Slav Petrov. 2019. Natural Questions: A Benchmark for Question  
 627 Answering Research. *Transactions of the Association for Computational Linguistics* 7 (2019),  
 628 452–466. [https://doi.org/10.1162/tacl\\_a\\_00276](https://doi.org/10.1162/tacl_a_00276) doi:10.1162/tacl\_a\_00276 Place :  
 629 Cambridge, MAPublisher : MIT Press.

630 Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao  
 631 Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. 2023. Efficient Memory  
 632 Management for Large Language Model Serving with PagedAttention.  
 633 <https://doi.org/10.48550/arXiv.2309.06180> doi:10.48550/arXiv.2309.06180 arXiv:2309.06180  
 634 [cs].

635 Yoonjoo Lee, Kyungjae Lee, Sunghyun Park, Dasol Hwang, Jaehyeon Kim, Hong-In Lee, and  
 636 Moontae Lee. 2023. QASA: Advanced Question Answering on Scientific Articles. In *Pro-  
 637 ceedings of the 40th International Conference on Machine Learning*. PMLR, 19036–19052.  
 638 <https://proceedings.mlr.press/v202/lee23n.html>

639

640 Dawei Li, Bohan Jiang, Liangjie Huang, Alimohammad Beigi, Chengshuai Zhao, Zhen Tan, Am-  
 641 ritra Bhattacharjee, Yuxuan Jiang, Canyu Chen, Tianhao Wu, Kai Shu, Lu Cheng, and Huan  
 642 Liu. 2025. From Generation to Judgment: Opportunities and Challenges of LLM-as-a-judge.  
 643 <https://doi.org/10.48550/arXiv.2411.16594> doi:10.48550/arXiv.2411.16594 arXiv:2411.16594  
 644 [cs].

645 Ruosen Li, Teerth Patel, and Xinya Du. 2024. PRD: Peer Rank  
 646 and Discussion Improve Large Language Model based Evaluations.  
 647 <https://doi.org/10.48550/arXiv.2307.02762> doi:10.48550/arXiv.2307.02762 arXiv:2307.02762  
 648 [cs].

648 Nelson Liu, Tianyi Zhang, and Percy Liang. 2023. Evaluating Verifiability in Gen-  
 649 erative Search Engines. In *Findings of the Association for Computational Linguistics:*  
 650 *EMNLP 2023*, Houda Bouamor, Juan Pino, and Kalika Bali (Eds.). Association for Com-  
 651 putational Linguistics, Singapore, 7001–7025. <https://doi.org/10.18653/v1/2023.findings-emnlp.467>

652

653 lotus. 2025. lotus-data/lotus. <https://github.com/lotus-data/lotus> original-date: 2024-07-  
 654 16T16:39:06Z.

655

656 Jannat Ara Meem, Muhammad Shihab Rashid, Yue Dong, and Vagelis Hristidis. 2024. PAT-  
 657 Questions: A Self-Updating Benchmark for Present-Anchored Temporal Question-Answering.  
 658 <https://doi.org/10.48550/arXiv.2402.11034> doi:10.48550/arXiv.2402.11034 arXiv:2402.11034  
 659 [cs].

660

661 Meta. 2025. meta-llama/Llama-4-Scout-17B-16E-Instruct · Hugging Face.  
 662 <https://huggingface.co/meta-llama/Llama-4-Scout-17B-16E-Instruct>

663 Grégoire Mialon, Clémentine Fourrier, Craig Swift, Thomas Wolf, Yann LeCun,  
 664 and Thomas Scialom. 2023. GAIA: a benchmark for General AI Assistants.  
 665 <https://doi.org/10.48550/arXiv.2311.12983> doi:10.48550/arXiv.2311.12983 arXiv:2311.12983  
 666 [cs].

667

668 Federico Nanni, Bhaskar Mitra, Matt Magnusson, and Laura Dietz. 2017. Benchmark for Complex  
 669 Answer Retrieval. <https://doi.org/10.48550/arXiv.1705.04803> doi:10.48550/arXiv.1705.04803  
 670 arXiv:1705.04803 [cs].

671 OpenAI. 2025a. Introducing deep research | OpenAI. <https://openai.com/index/introducing-deep-research/>

672

673 OpenAI. 2025b. Model - OpenAI API. <https://platform.openai.com>

674

675 OpenAlex. 2025. OpenAlex: The open catalog to the global research system | OpenAlex.  
 676 <https://openalex.org/>

677

678 Jie Ouyang, Tingyue Pan, Mingyue Cheng, Ruiran Yan, Yucong Luo, Jiaying Lin, and Qi Liu. 2025. HoH: A Dynamic Benchmark for Evaluating the Impact of Outdated Information on Retrieval-Augmented Generation.  
 679 <https://doi.org/10.48550/arXiv.2503.04800> doi:10.48550/arXiv.2503.04800 arXiv:2503.04800  
 680 [cs].

681

682 Liana Patel, Siddharth Jha, Melissa Pan, Harshit Gupta, Parth Asawa, Carlos Guestrin, and Matei Zaharia. 2025. Semantic Operators: A Declarative Model for Rich, AI-based Data Processing.  
 683 <https://doi.org/10.48550/arXiv.2407.11418> doi:10.48550/arXiv.2407.11418 arXiv:2407.11418  
 684 [cs].

685

686 Perplexity. 2025. Introducing Perplexity Deep Research.  
 687 <https://www.perplexity.ai/hub/blog/introducing-perplexity-deep-research>

688

689 Ronak Pradeep, Nandan Thakur, Shivani Upadhyay, Daniel Campos, Nick Craswell, and Jimmy Lin.  
 690 2025. The Great Nugget Recall: Automating Fact Extraction and RAG Evaluation with Large  
 691 Language Models. <https://doi.org/10.48550/arXiv.2504.15068> doi:10.48550/arXiv.2504.15068  
 692 arXiv:2504.15068 [cs].

693

694 Qwen, An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan  
 695 Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang,  
 696 Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin  
 697 Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao  
 698 Li, Tianyi Tang, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang  
 699 Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. 2025. Qwen2.5  
 700 Technical Report. <https://doi.org/10.48550/arXiv.2412.15115> doi:10.48550/arXiv.2412.15115  
 701 arXiv:2412.15115 [cs].

702 Hossein A. Rahmani, Clemencia Siro, Mohammad Aliannejadi, Nick Craswell, Charles L. A.  
 703 Clarke, Guglielmo Faggioli, Bhaskar Mitra, Paul Thomas, and Emine Yilmaz. 2024a. LLM4Eval:  
 704 Large Language Model for Evaluation in IR. In *Proceedings of the 47th International ACM SIGIR  
 705 Conference on Research and Development in Information Retrieval*. ACM, Washington DC USA,  
 706 3040–3043. <https://doi.org/10.1145/3626772.3657992>doi:10.1145/3626772.3657992

707 Hossein A. Rahmani, Emine Yilmaz, Nick Craswell, Bhaskar Mitra, Paul  
 708 Thomas, Charles L. A. Clarke, Mohammad Aliannejadi, Clemencia Siro, and  
 709 Guglielmo Faggioli. 2024b. LLMJudge: LLMs for Relevance Judgments.  
 710 <https://doi.org/10.48550/arXiv.2408.08896>doi:10.48550/arXiv.2408.08896 arXiv:2408.08896  
 711 [cs].

712 Yijia Shao, Yucheng Jiang, Theodore A. Kanell, Peter Xu, Omar Khattab, and Monica S. Lam.  
 713 2024. Assisting in Writing Wikipedia-like Articles From Scratch with Large Language Models.  
 714 <https://doi.org/10.48550/arXiv.2402.14207>doi:10.48550/arXiv.2402.14207 arXiv:2402.14207  
 715 [cs].

716 Luca Soldaini, Rodney Kinney, Akshita Bhagia, Dustin Schwenk, David Atkinson, Russell Au-  
 717 thur, Ben Bogin, Khyathi Chandu, Jennifer Dumas, Yanai Elazar, Valentin Hofmann, Ananya  
 718 Jha, Sachin Kumar, Li Lucy, Xinxixi Lyu, Nathan Lambert, Ian Magnusson, Jacob Morrison,  
 719 Niklas Muennighoff, Aakanksha Naik, Crystal Nam, Matthew Peters, Abhilasha Ravichander,  
 720 Kyle Richardson, Zejiang Shen, Emma Strubell, Nishant Subramani, Oyvind Tafjord, Evan  
 721 Walsh, Luke Zettlemoyer, Noah Smith, Hannaneh Hajishirzi, Iz Beltagy, Dirk Groeneveld, Jesse  
 722 Dodge, and Kyle Lo. 2024. Dolma: an Open Corpus of Three Trillion Tokens for Language  
 723 Model Pretraining Research. In *Proceedings of the 62nd Annual Meeting of the Association for  
 724 Computational Linguistics (Volume 1: Long Papers)*, Lun-Wei Ku, Andre Martins, and Vivek  
 725 Srikumar (Eds.). Association for Computational Linguistics, Bangkok, Thailand, 15725–15788.  
 726 <https://doi.org/10.18653/v1/2024.acl-long.840>doi:10.18653/v1/2024.acl-long.840

727 Weihang Su, Anzhe Xie, Qingyao Ai, Jianming Long, Jiaxin Mao, Ziyi Ye,  
 728 and Yiqun Liu. 2025. Benchmarking Computer Science Survey Generation.  
 729 <https://doi.org/10.48550/arXiv.2508.15658>doi:10.48550/arXiv.2508.15658 arXiv:2508.15658  
 730 [cs].

731 Nandan Thakur, Jimmy Lin, Sam Havens, Michael Carbin, Omar Khattab, and Andrew Droz-  
 732 dov. 2025. FreshStack: Building Realistic Benchmarks for Evaluating Retrieval on Tech-  
 733 nical Documents. <https://doi.org/10.48550/arXiv.2504.13128>doi:10.48550/arXiv.2504.13128  
 734 arXiv:2504.13128 [cs].

735 Nandan Thakur, Nils Reimers, Andreas Rücklé, Abhishek Srivastava, and Iryna Gurevych. 2021.  
 736 BEIR: A Heterogenous Benchmark for Zero-shot Evaluation of Information Retrieval Models.  
 737 <https://doi.org/10.48550/arXiv.2104.08663>doi:10.48550/arXiv.2104.08663 arXiv:2104.08663  
 738 [cs].

739 Paul Thomas, Seth Spielman, Nick Craswell, and Bhaskar Mitra. 2024.  
 740 Large language models can accurately predict searcher preferences.  
 741 <https://doi.org/10.48550/arXiv.2309.10621>doi:10.48550/arXiv.2309.10621 arXiv:2309.10621  
 742 [cs].

743 Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal.  
 744 2022. MuSiQue: Multihop Questions via Single-hop Question Composition.  
 745 <https://doi.org/10.48550/arXiv.2108.00573>doi:10.48550/arXiv.2108.00573 arXiv:2108.00573  
 746 [cs].

747 Shivani Upadhyay, Ronak Pradeep, Nandan Thakur, Daniel Campos, Nick Craswell,  
 748 Ian Soboroff, Hoa Trang Dang, and Jimmy Lin. 2024a. A Large-Scale Study  
 749 of Relevance Assessments with Large Language Models: An Initial Look.  
 750 <https://doi.org/10.48550/arXiv.2411.08275>doi:10.48550/arXiv.2411.08275 arXiv:2411.08275  
 751 [cs].

752 Shivani Upadhyay, Ronak Pradeep, Nandan Thakur, Nick Craswell, and Jimmy Lin. 2024b.  
 753 UMBRELA: UMbrela is the (Open-Source Reproduction of the) Bing RELevance Assessor.

756 https://doi.org/10.48550/arXiv.2406.06519doi:10.48550/arXiv.2406.06519 arXiv:2406.06519  
 757 [cs].  
 758

759 Ellen M. Voorhees. 2009. I Come Not To Bury Cranfield, but to Praise It. *NIST* (Oct. 2009).  
 760 https://www.nist.gov/publications/i-come-not-bury-cranfield-praise-it Last Modified: 2017-02-  
 761 19T20:02-05:00 Publisher: Ellen M. Voorhees.

762 David Wadden, Shanchuan Lin, Kyle Lo, Lucy Lu Wang, Madeleine van Zuylen, Arman  
 763 Cohan, and Hannaneh Hajishirzi. 2020. Fact or Fiction: Verifying Scientific Claims. In  
 764 *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing* (EMNLP),  
 765 Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu (Eds.). Association  
 766 for Computational Linguistics, Online, 7534–7550. https://doi.org/10.18653/v1/2020.emnlp-  
 767 main.609doi:10.18653/v1/2020.emnlp-main.609

768 Haiyuan Wan, Chen Yang, Junchi Yu, Meiqi Tu, Jiaxuan Lu, Di Yu, Jianbao Cao, Ben Gao,  
 769 Jiaqing Xie, Aoran Wang, Wenlong Zhang, Philip Torr, and Dongzhan Zhou. 2025. Deep-  
 770 Research Arena: The First Exam of LLMs' Research Abilities via Seminar-Grounded Tasks.  
 771 https://doi.org/10.48550/arXiv.2509.01396doi:10.48550/arXiv.2509.01396 arXiv:2509.01396  
 772 [cs].

773 Jason Wei, Nguyen Karina, Hyung Won Chung, Yunxin Joy Jiao, Spencer Papay, Amelia  
 774 Glaese, John Schulman, and William Fedus. 2024. Measuring short-form factuality in large  
 775 language models. https://doi.org/10.48550/arXiv.2411.04368doi:10.48550/arXiv.2411.04368  
 776 arXiv:2411.04368 [cs].

777 Jason Wei, Zhiqing Sun, Spencer Papay, Scott McKinney, Jeffrey Han, Isa Fulford,  
 778 Hyung Won Chung, Alex Tachard Passos, William Fedus, and Amelia Glaese.  
 779 2025. BrowseComp: A Simple Yet Challenging Benchmark for Browsing Agents.  
 780 https://doi.org/10.48550/arXiv.2504.12516doi:10.48550/arXiv.2504.12516 arXiv:2504.12516  
 781 [cs].  
 782

783 Theodora Worledge, Tatsunori Hashimoto, and Carlos Guestrin. 2024. The Extractive-  
 784 Abstractive Spectrum: Uncovering Verifiability Trade-offs in LLM Generations.  
 785 https://doi.org/10.48550/arXiv.2411.17375doi:10.48550/arXiv.2411.17375 arXiv:2411.17375  
 786 [cs].

787 Jialong Wu, Wenbiao Yin, Yong Jiang, Zhenglin Wang, Zekun Xi, Runnan Fang, Linhai Zhang, Yu-  
 788 lan He, Deyu Zhou, Pengjun Xie, and Fei Huang. 2025. WebWalker: Benchmarking LLMs  
 789 in Web Traversal. https://doi.org/10.48550/arXiv.2501.07572doi:10.48550/arXiv.2501.07572  
 790 arXiv:2501.07572 [cs].

791 xAI. 2025. Grok 3 Beta — The Age of Reasoning Agents | xAI. https://x.ai/news/grok-3  
 792

793 Tianze Xu, Pengrui Lu, Lyumanshan Ye, Xiangkun Hu, and Pengfei Liu. 2025. Re-  
 794 searcherBench: Evaluating Deep AI Research Systems on the Frontiers of Scientific Inquiry.  
 795 https://doi.org/10.48550/arXiv.2507.16280doi:10.48550/arXiv.2507.16280 arXiv:2507.16280  
 796 [cs].

797 Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. Cohen, Ruslan Salakhutdinov,  
 798 and Christopher D. Manning. 2018. HotpotQA: A Dataset for Diverse, Explainable Multi-hop  
 799 Question Answering. https://doi.org/10.48550/arXiv.1809.09600doi:10.48550/arXiv.1809.09600  
 800 arXiv:1809.09600 [cs].

801 you.com. 2025. Su-Sea/ydc-deep-research-evals: you.com's framework for evaluating deep research  
 802 systems. https://github.com/Su-Sea/ydc-deep-research-evals

803 Haozhen Zhang, Tao Feng, Pengrui Han, and Jiaxuan You. 2024b. AcademicEval: Live Long-  
 804 Context LLM Benchmark. (Oct. 2024). https://openreview.net/forum?id=iRYExPKnxm

805 Jiajie Zhang, Yushi Bai, Xin Lv, Wanjun Gu, Danqing Liu, Minhao Zou, Shulin  
 806 Cao, Lei Hou, Yuxiao Dong, Ling Feng, and Juanzi Li. 2024a. LongCite:  
 807 Enabling LLMs to Generate Fine-grained Citations in Long-context QA.  
 808 https://doi.org/10.48550/arXiv.2409.02897doi:10.48550/arXiv.2409.02897 arXiv:2409.02897  
 809 [cs].

810 Yuxiang Zheng, Dayuan Fu, Xiangkun Hu, Xiaojie Cai, Lyumanshan Ye, Pengrui Lu, and Pengfei  
811 Liu. 2025. DeepResearcher: Scaling Deep Research via Reinforcement Learning in Real-world  
812 Environments. <https://arxiv.org/abs/2504.03160v4>

813 Yuxiang Zheng, Shichao Sun, Lin Qiu, Dongyu Ru, Cheng Jiayang, Xuefeng Li, Jifan Lin,  
814 Binjie Wang, Yun Luo, Renjie Pan, Yang Xu, Qingkai Min, Zizhao Zhang, Yiwen Wang,  
815 Wenjie Li, and Pengfei Liu. 2024. OpenResearcher: Unleashing AI for Accelerated Scien-  
816 tific Research. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Lan-*  
817 *guage Processing: System Demonstrations*, Delia Irazu Hernandez Farias, Tom Hope, and  
818 Manling Li (Eds.). Association for Computational Linguistics, Miami, Florida, USA, 209–218.  
819 <https://doi.org/10.18653/v1/2024.emnlp-demo.22>  
820  
821  
822  
823  
824  
825  
826  
827  
828  
829  
830  
831  
832  
833  
834  
835  
836  
837  
838  
839  
840  
841  
842  
843  
844  
845  
846  
847  
848  
849  
850  
851  
852  
853  
854  
855  
856  
857  
858  
859  
860  
861  
862  
863

864  
865  
866  
867  
868  
869  
870  
871  
872  
873  
874  
875  
876  
877  
878  
879  
880  
881  
882  
883  
884  
885  
886  
887  
888  
889  
890  
891  
892  
893  
894  
895  
896  
897  
898  
899  
900  
901  
902  
903  
904  
905  
906  
907  
908  
909  
910  
911  
912  
913  
914  
915  
916  
917  
8 APPENDIX

| FIELD NAME                       | DESCRIPTION                                       | EXAMPLE  | FIELD NAME             | DESCRIPTION  | EXAMPLE   |
|----------------------------------|---|--|------------------------|--|---|
| arxiv_id                         | ArXiv paper ID                                    | 2506.02838v1   | Parent Paper           | <i>Info about the paper in which the citation appears.</i> |   |
| title                            | Paper title                                       | TaxAgent: How Large Language Model Designs Fis...    | parent_paper_title     | Title of the parent paper.                                 | TaxAgent: How Large Language Model Designs Fis...                                   |
| authors                          | Comma-separated list of authors                   | Jizhou Wang, Xiaodan Fang, Lei Huang, Yongfeng...    | parent_paper_arxiv_id  | ArXiv ID of the parent paper.                              | 2506.02838v1  |
| abstract                         | Paper abstract from ArXiv                         | Economic inequality is a global challenge, int...    | Cited Paper            | <i>Details from the reference entry.</i>                   |   |
| categories                       | ArXiv categories                                  | cs.AI, econ.GN, q-fin.EC, 12.11, 16.5                | citation_shorthand     | Citation key in the bibliography.                          | NBERw21340  |
| published_date                   | First Publication date                            | 2025-06-03T13:06:19+00:00                            | cited_paper_title      | Title as listed in the reference list.                     | Effective Policy for Reducing Inequality? The ...                                   |
| clean_latex_related_works        | Related Works section derived from LATEX of paper | 'subsection{Traditional Tax Systems}' inProgress ... | cited_paper_arxiv_link | ArXiv link if provided.                                    | NaN   |
| <b>Bibliographic Data</b>        |   |  |                        |  |   |
|                                  |   |  | bib_paper_authors      | Authors from external metadata.                            | Hoynes, Hilary W and Patel, Ankur J   |
|                                  |   |  | bib_paper_year         | Publication year   | 2015  |
|                                  |   |  | bib_paper_month        | Publication month  | July  |
|                                  |   |  | bib_paper_url          | URL from bibliographic records.                            | <a href="http://www.nber.org/papers/w21340">http://www.nber.org/papers/w21340</a>   |
|                                  |   |  | bib_paper_doi          | DOI from external metadata.                                | 10.3386/w21340  |
|                                  |   |  | bib_paper_journal      | Journal or series name.                                    | NaN   |
|                                  |   |  | original_title         | Official title from bibliographic databases.               | Effective Policy for Reducing Inequality? The ...                                   |
| <b>Search Results (Verified)</b> |   |  |                        |  |   |
|                                  |   |  | search_res_title       | Title from the search result.                              | Effective Policy for Reducing Inequality? The ...                                   |
|                                  |   |  | search_res_url         | URL from the search result.                                | <a href="https://www.nber.org/papers/w21340">https://www.nber.org/papers/w21340</a> |
|                                  |   |  | search_res_content     | Abstract snippet from the search result page.              | We use a quasi-experiment approach, using vari...                                   |

Figure 3: DeepScholar-bench dataset schema.

## 8.1 DEEPSCHOLAR-BENCH DATASET

We provide a detailed overview and schema of the DeepScholar-bench dataset in Figure 3. In addition, Table 5 includes the papers that are included in DeepScholar-June-2025.

## 8.2 OVERVIEW OF BASELINES AND EXPERIMENTAL SETUP

We briefly overview all of the baseline systems we evaluate, which includes recent state-of-the-art generative research systems as well as DeepScholar-ref on DeepScholar-bench. We benchmark open-source research systems, including DeepResearcher Zheng et al. (2025), STORM Shao et al. (2024) and OpenScholar Asai et al. (2024), search agents, with Llama-4-Scout-17B-16E-Instruct Meta (2025), GPT-4.1-2025-04-14 OpenAI (2025b), o3-2025-04-16 OpenAI (2025b), Claude-opus-4-20250514 Anthropic (2025b), and Gemini-2.5-pro Gemini (2025b) models, OpenAI’s o3-deep-research OpenAI (2025b), and DeepScholar-ref.

We report results using GPT-4.1-2025-04-14 (OpenAI, 2025b) as the judge for Nugget Coverage, and a GPT-4o-2024-08-06 (OpenAI, 2025b) judge for Organization, Relevance Rate, Reference Coverage, Citation Precision and Claim Coverage. We report the Organization score as a win rate including ties, we report the strict all score for Nugget Coverage, and we report Claim Coverage with a window size of  $w = 1$ . For all Retrieval Quality metrics, we consider the retrieved set of each given report as the set of any valid ArXiv links found within the report. To measure Document Importance, we use the OpenAlex (OpenAlex, 2025) API to recover citation information. For each metric, we report an average over all reports.

910  
911  
8.2.1 OPEN-SOURCE RESEARCH SYSTEMS

We evaluate three state-of-the-art open-source systems, DeepResearcher Zheng et al. (2025), STORM Shao et al. (2024) and OpenScholar Asai et al. (2024). For each, we run these systems using the Llama-4-Scout-17B-16E-Instruct model Meta (2025), which we serve with 4 A100 GPUs using vLLM Kwon et al. (2023).

DeepResearcher Zheng et al. (2025) leverages trained agents to navigate, browse and synthesize information from the web. To train an agent, this work uses end-to-end reinforcement learning and trains Qwen2.5-7B-Instruct Qwen et al. (2025). In our benchmarks,

918 Table 5: Title and ArXiv IDs of papers included in DeepScholar-June-2025.  
919

| #  | ArXiv ID     | Title  |
|----|--------------|--|
| 0  | 2506.02838v1 | TaxAgent: How Large Language Model Designs Fiscal Policy   |
| 1  | 2506.02634v1 | KVCache Cache in the Wild: Characterizing and Optimizing KVCache Cache at a Large Cloud Provider                       |
| 2  | 2506.00958v1 | Speaking Beyond Language: A Large-Scale Multimodal Dataset for Learning Nonverbal Cues from Video-Grounded Dialogues   |
| 3  | 2506.00832v1 | Counterfactual Activation Editing for Post-hoc Prosody and Mispronunciation Correction in TTS Models                   |
| 4  | 2506.00418v1 | Dual Debiasing for Noisy In-Context Learning for Text Generation   |
| 5  | 2505.24754v1 | Don't Reinvent the Wheel: Efficient Instruction-Following Text Embedding based on Guided Space Transformation          |
| 6  | 2505.24575v1 | NexusSum: Hierarchical LLM Agents for Long-Form Narrative Summarization  |
| 7  | 2506.00085v1 | COSMIC: Generalized Refusal Direction Identification in LLM Activations  |
| 8  | 2505.23996v1 | Is Your Model Fairly Certain? Uncertainty-Aware Fairness Evaluation for LLMs   |
| 9  | 2505.23353v1 | Synthetic Generation and Latent Projection Denoising of Rim Lesions in Multiple Sclerosis                              |
| 10 | 2505.22757v1 | Pre-Training Curriculum for Multi-Token Prediction in Language Models  |
| 11 | 2506.02853v1 | Learning Pyramid-structured Long-range Dependencies for 3D Human Pose Estimation                                       |
| 12 | 2506.02547v1 | Probabilistic Online Event Downsampling  |
| 13 | 2506.01071v1 | Aligned Contrastive Loss for Long-Tailed Recognition   |
| 14 | 2506.01037v1 | Self-supervised ControlNet with Spatio-Temporal Mamba for Real-world Video Super-resolution                            |
| 15 | 2506.00434v1 | Efficient 3D Brain Tumor Segmentation with Axial-Coronal-Sagittal Embedding  |
| 16 | 2506.00333v1 | Test-time Vocabulary Adaptation for Language-driven Object Detection   |
| 17 | 2505.24443v1 | Diversify and Conquer: Open-set Disagreement for Robust Semi-supervised Learning with Outliers                         |
| 18 | 2505.24334v1 | KairosAD: A SAM-Based Model for Industrial Anomaly Detection on Embedded Devices                                       |
| 19 | 2505.23290v1 | Wav2Sem: Plug-and-Play Audio Semantic Decoupling for 3D Speech-Driven Facial Animation                                 |
| 20 | 2505.23180v1 | Proximal Algorithm Unrolling: Flexible and Efficient Reconstruction Networks for Single-Pixel Imaging                  |
| 21 | 2505.22616v1 | PS4PRO: Pixel-to-pixel Supervision for Photorealistic Rendering and Optimization                                       |
| 22 | 2505.22458v1 | Universal Domain Adaptation for Semantic Segmentation  |
| 23 | 2505.22427v1 | RC-AutoCalib: An End-to-End Radar-Camera Automatic Calibration Network   |
| 24 | 2505.22167v1 | Q-VDT: Towards Accurate Quantization and Distillation of Video-Generation Diffusion Transformers                       |
| 25 | 2505.22552v1 | ClaimPKG: Enhancing Claim Verification via Pseudo-Subgraph Generation with Lightweight Specialized LLM                 |
| 26 | 2504.21752v1 | VDDP: Verifiable Distributed Differential Privacy under the Client-Server-Verifier Setup                               |
| 27 | 2504.21282v1 | Birdie: Natural Language-Driven Table Discovery Using Differentiable Search Index                                      |
| 28 | 2504.17448v1 | CHASE: Client Heterogeneity-Aware Data Selection for Effective Federated Active Learning                               |
| 29 | 2504.14861v1 | Stitching Inner Product and Euclidean Metrics for Topology-aware Maximum Inner Product Search                          |
| 30 | 2504.06975v1 | AWDIT: An Optimal Weak Database Isolation Tester   |
| 31 | 2506.01833v1 | SPACE: Your Genomic Profile Predictor is a Powerful DNA Foundation Model   |
| 32 | 2506.00382v1 | Spectral Insights into Data-Oblivious Critical Layers in Large Language Models   |
| 33 | 2506.00205v1 | Unlocking the Power of Rehearsal in Continual Learning: A Theoretical Perspective                                      |
| 34 | 2505.24835v1 | Timing is important: Risk-aware Fund Allocation based on Time-Series Forecasting                                       |
| 35 | 2505.24203v1 | Aligning Protein Conformation Ensemble Generation with Physical Feedback   |
| 36 | 2506.02847v1 | CLONE: Customizing LLMs for Efficient Latency-Aware Inference at the Edge  |
| 37 | 2505.22194v1 | Refining Datapath for Microscaling ViTs  |
| 38 | 2505.11554v1 | Multi-Objective Memory Bandwidth Regulation and Cache Partitioning for Multicore Real-Time Systems                     |
| 39 | 2505.08071v1 | NMP-PaK: Near-Memory Processing Acceleration of Scalable De Novo Genome Assembly                                       |
| 40 | 2504.06211v1 | Need for zkSpeed: Accelerating HyperPlonk for Zero-Knowledge Proofs  |
| 41 | 2504.19283v1 | Efficient Serverless Cold Start: Reducing Library Loading Overhead by Profile-guided Optimization                      |
| 42 | 2504.11007v1 | Kubernetes in the Cloud vs. Bare Metal: A Comparative Study of Network Costs   |
| 43 | 2504.09307v1 | Lumos: Efficient Performance Modeling and Estimation for Large-scale LLM Training                                      |
| 44 | 2506.02750v1 | Learning Binarized Representations with Pseudo-positive Sample Enhancement for Efficient Graph Collaborative Filtering |
| 45 | 2505.23452v1 | What About Emotions? Guiding Fine-Grained Emotion Extraction from Mobile App Reviews                                   |
| 46 | 2505.21811v1 | Revisiting Self-attention for Cross-domain Sequential Recommendation   |
| 47 | 2505.20227v1 | Measure Domain's Gap: A Similar Domain Selection Principle for Multi-Domain Recommendation                             |
| 48 | 2505.19356v1 | Optimized Text Embedding Models and Benchmarks for Amharic Passage Retrieval   |
| 49 | 2505.19307v1 | Aligning Web Query Generation with Ranking Objectives via Direct Preference Optimization                               |
| 50 | 2505.17507v1 | Benchmarking Recommendation, Classification, and Tracing Based on Hugging Face Knowledge Graph                         |
| 51 | 2505.12791v1 | Unlearning for Federated Online Learning to Rank: A Reproducibility Study  |
| 52 | 2505.07166v1 | Pre-training vs. Fine-tuning: A Reproducibility Study on Dense Retrieval Knowledge Acquisition                         |
| 53 | 2505.03484v1 | STAR-Rec: Making Peace with Length Variance and Pattern Diversity in Sequential Recommendation                         |
| 54 | 2505.00552v1 | Graph Spectral Filtering with Chebyshev Interpolation for Recommendation   |
| 55 | 2504.20458v1 | Search-Based Interaction For Conversation Recommendation via Generative Reward Model Based Simulated User              |
| 56 | 2504.18383v1 | Bridge the Domains: Large Language Models Enhanced Cross-domain Sequential Recommendation                              |
| 57 | 2504.17519v1 | Replication and Exploration of Generative Retrieval over Dynamic Corpora   |
| 58 | 2504.15849v1 | NLCTables: A Dataset for Marrying Natural Language Conditions with Table Discovery                                     |
| 59 | 2504.14991v1 | Understanding Accuracy-Fairness Trade-offs in Re-ranking through Elasticity in Economics                               |
| 60 | 2504.14243v1 | Unconstrained Monotonic Calibration of Predictions in Deep Ranking Systems   |
| 61 | 2504.12900v1 | FashionDPO: Fine-tune Fashion Outfit Generation Model using Direct Preference Optimization                             |
| 62 | 2504.09935v1 | Constrained Auto-Regressive Decoding Constrains Generative Retrieval   |

972 we evaluate DeepResearcher using both the released, trained model from the authors, and using Llama-4-Scout-17B-16E-Instruct model Meta (2025) as the core LLM. We report the better performing baseline of these two, which we find in our experiments to be the Llama-4-Scout-17B-16E-Instruct backbone.

973 **STORM** Shao et al. (2024) studies the problem of how to apply LLMs to write grounded, organized  
 974 long-form articles (e.g., Wikipedia articles) from scratch. The system involves a pre-writing stage  
 975 that discovers diverse research perspectives on a topic by stimulating conversations between multiple  
 976 agents and leveraging web documents.

977 OpenScholar Asai et al. (2024) builds a specialized retrieval-augmented LLM system for literature  
 978 synthesis and scientific queries. This method includes a trained retriever from the pre-indexed  
 979 peS2o Soldaini et al. (2024) corpus, consisting of 45 million open-access academic papers up until  
 980 October 2024, as an initial retrieval source before using web search. In our experiments, we  
 981 benchmark the system using this pre-indexed corpus and limit web search to the ArXiv API.

### 982 8.2.2 SEARCH AGENTS

983 We evaluate the following models: Llama-4-Scout-17B-16E-Instruct Meta  
 984 (2025), GPT-4.1-2025-04-14 OpenAI (2025b), o3-2025-04-16 OpenAI (2025b),  
 985 Claude-opus-4-20250514 Anthropic (2025b), and Gemini-2.5-pro Gemini (2025b). We  
 986 augment each with search capabilities to ArXiv arxiv (2025), and use the popular ODS  
 987 framework Alzubi et al. (2025) to allow the LLM to make tool calls to the search API.

### 988 8.2.3 COMMERCIAL SYSTEMS.

989 We focus our evaluation of commercial generative research synthesis systems on OpenAI’s o3-deep-  
 990 research OpenAI (2025b), which provides a public API allowing for our evaluation.

### 991 8.2.4 DEEPSCHOLAR-REF

992 Similar to our evaluation of search agent we evaluate DeepScholar-ref with the following models:  
 993 Llama-4-Scout-17B-16E-Instruct Meta (2025), GPT-4.1-2025-04-14 OpenAI (2025b),  
 994 o3-2025-04-16 OpenAI (2025b), Claude-opus-4-20250514 Anthropic (2025b),  
 995 and Gemini-2.5-pro Gemini (2025b). For each of these baselines, we also use the same or a  
 996 weaker model, either Llama-4 or GPT-4.1, to perform semantic filtering and top-k operators. We  
 997 limit the method to two rounds of search, each with at most 2 queries.

## 1000 8.3 ADDITIONAL EXPERIMENTAL RESULTS

### 1001 8.3.1 METADATA STATISTICS

1002 We provide metadata statistics characterizing the generated reports of each benchmarked method in  
 1003 Table 6, as well as statistics related to our evaluation metrics in Table 7.

### 1004 8.3.2 RESULTS ON DEEPSCHOLAR-NOV-2025

1005 To study how our benchmark behaves under both domain and temporal shifts in the live arXiv API,  
 1006 we instantiate a second benchmark slice, in addition to DeepScholar-June-2025 (Section 3 which  
 1007 is built from 63 Computer Science papers on arXiv to study domain coverage and robustness of  
 1008 DeepScholar-bench.

1009 DEEPSCHOLAR-Nov-2025 contains 200 queries sampled from more than 75 distinct arXiv subject  
 1010 areas, spanning Computer Science, Physics, Quantitative Biology, Economics, and Quantitative  
 1011 Finance. We evaluate a subset of high-performing systems to check whether our conclusions remain  
 1012 stable: DeepScholar-ref instantiated with Llama-4-Scout-17B-16E and with GPT-4.1+o3, and the  
 1013 Search Agent baseline instantiated with the same model configurations.

1014 Table 8 reports the resulting scores across all seven metrics. Overall, we observe patterns that are  
 1015 consistent with our main results on DeepScholar-June-2025 (Table 2). For both DeepScholar-ref  
 1016 and the Search Agent baseline, the o3-based variants substantially outperform their Llama-4-Scout

Table 6: Report Statistics.

|                                       | Chars | Report Length |           | Citations     |                    | Latency (s) | Cost              |
|---------------------------------------|-------|---------------|-----------|---------------|--------------------|-------------|-------------------|
|                                       |       | Words         | Sentences | # Unique Refs | # Inline Citations |             | Dollar Cost (USD) |
| <i>Human-written Exemplars</i>        |       |               |           |               |                    |             |                   |
| Human-written Exemplars               | 4381  | 497           | 28        | 23            | 27                 | N/A         | N/A               |
| <i>Open Source Research Systems</i>   |       |               |           |               |                    |             |                   |
| DeepResearcher (Llama-4)              | 2573  | 319           | 35        | 8             | 7                  | 31          | 0.00              |
| STORM (Llama-4)                       | 2766  | 381           | 31        | 18            | 21                 | 162         | 0.00              |
| OpenScholar (Llama-4)                 | 3513  | 483           | 26        | 9             | 19                 | 87          | 0.00              |
| <i>Search Agents</i>                  |       |               |           |               |                    |             |                   |
| Search Agent (Llama-4)                | 1968  | 258           | 16        | 9             | 5                  | 20          | 0.00              |
| Search Agent (GPT-4.1)                | 3168  | 404           | 16        | 10            | 61                 | 39          | 0.07              |
| Search Agent (o3)                     | 3844  | 501           | 24        | 11            | 16                 | 263         | 0.15              |
| Search Agent (Claude)                 | 3977  | 499           | 27        | 13            | 8                  | 147         | 1.36              |
| Search Agent (Gemini)                 | 2810  | 395           | 19        | 6             | 8                  | 442         | 0.11              |
| <i>Commercial Systems</i>             |       |               |           |               |                    |             |                   |
| OpenAI DeepResearch                   | 6577  | 864           | 74        | 17            | 6                  | 630         | 5.02              |
| <i>DeepScholar Reference Pipeline</i> |       |               |           |               |                    |             |                   |
| DeepScholar-ref (Llama-4)             | 3499  | 360           | 53        | 19            | 35                 | 313         | 0.00              |
| DeepScholar-ref (GPT-4.1)             | 7863  | 735           | 115       | 20            | 89                 | 234         | 1.66              |
| DeepScholar-ref (GPT-4.1, o3)         | 5726  | 617           | 70        | 17            | 42                 | 276         | 1.15              |
| DeepScholar-ref (GPT-4.1, Claude)     | 5855  | 618           | 72        | 17            | 40                 | 334         | 1.23              |
| DeepScholar-ref (GPT-4.1, Gemini)     | 5623  | 570           | 86        | 24            | 63                 | 349         | 1.29              |

Table 7: Statistics Related to Evaluation Metrics.

|  | Avg. value over human-written exemplars | Relevant Metric        |
|--|---|------------------------|
| # Important References from ArXiv.org<br>Median number of citations per reference from ArXiv.org | 11.47<br>647.5                          | Ref. Cov.<br>Doc. Imp. |

counterparts on Organization, Nugget Coverage, Relevance Rate, Citation Precision, and Claim Coverage, while Reference Coverage and Document Importance remain broadly low across baselines. These trends mirror the relative ranking and qualitative gaps seen in our main benchmark slice, suggesting that our conclusions from DeepScholar-June-2025 generalize robust beyond queries related to Computer Science.

We emphasize that DeepScholar-Bench is defined by an automated data-curation and evaluation pipeline rather than a single fixed dataset. Instantiating new slices such as DEEPSCHOLAR-NOV-2025 only requires specifying a set of query papers and a date range; the pipeline then automatically constructs the corresponding benchmark and produces scores under the same evaluation protocol. This design allows practitioners to easily create additional domain- or time-specific evaluations while remaining comparable to our core results.

### 8.3.3 ABLATION STUDY: UNDERSTANDING THE PERFORMANCE IMPACT OF THE CHOSEN QUERY

Our main experiments use the paper abstract as the query description  $d$  (Section 3.2). A natural question is whether our conclusions depend on this particular choice of query formulation. To assess this, we run an ablation in which we replace the abstract with two alternative, realistic user queries: (i) a two-sentence summary of the paper’s key idea (KEY IDEA) and (ii) a single research question describing the paper’s main goal (RQ). For each paper, we prompt an LLM to convert the abstract into these two alternative query formulations. We then re-run the full benchmark for each query version and compute system-level scores for all metrics across our main baselines.

Table 9 reports Pearson correlations between system-level scores obtained under different query formulations (rows) for each metric (columns) with statistical significance testing based on a permutation-based paired test ( $p < 0.05$ ). Overall, we observe very strong agreement across query types: correlations are typically above 0.95 for Organization, Nugget Coverage, Reference Coverage, Document Importance, and Claim Coverage, and above 0.77 for Coverage Relevance Rate in

1080 Table 8: Performance of selected systems on DEEPSCHOLAR-NOV-2025 (200 queries across >75  
 1081 arXiv subject areas). We report the same metrics as in Table 2.

| Model                           | Org.  | Nug. Cov. | Rel. Rate | Ref Cov. | Doc Imp. | Cite-P | Claim Cov. |
|---------------------------------|-------|-----------|-----------|----------|----------|--------|------------|
| DeepScholar-ref (Llama-4-Scout) | 0.120 | 0.358     | 0.395     | 0.072    | 0.082    | 0.178  | 0.581      |
| DeepScholar-ref (GPT-4.1 + o3)  | 0.578 | 0.479     | 0.568     | 0.087    | 0.056    | 0.563  | 0.578      |
| Search Agent (Llama-4-Scout)    | 0.108 | 0.252     | 0.179     | 0.034    | 0.079    | 0.189  | 0.314      |
| Search Agent (o3)               | 0.608 | 0.480     | 0.623     | 0.078    | 0.034    | 0.475  | 0.452      |

1088 Table 9: Pearson correlations between system-level scores under different query formulations (abstract,  
 1089 KEY IDEA, and RQ). Each entry is computed over the scores of our main baselines. Correlations marked with \* are statistically significant under a permutation-based paired test with threshold  
 1090  $p < 0.05$ .

| Query pair            | Org.   | Nug.   | Cov. | Rel. Rate | Ref Cov | Doc Imp | Cite-P | Claim Cov ( $w=1$ ) |
|-----------------------|--------|--------|------|-----------|---------|---------|--------|---------------------|
| Abstract vs. Key Idea | 0.997* | 0.980* |      | 0.979*    | 0.992*  | 0.992   | 0.589  | 0.841*              |
| Abstract vs. RQ       | 0.988* | 0.979* |      | 0.772*    | 0.951*  | 0.707   | 0.766* | 0.877*              |
| Key Idea vs. RQ       | 0.991* | 0.983* |      | 0.836*    | 0.958*  | 0.766*  | 0.942* | 0.981*              |

1093 all cases. All correlations between KEY IDEA and RQ are statistically significant, indicating that  
 1094 more natural, user-facing query formulations yield highly consistent system rankings.

1095 The main sensitivity to query formulation arises for Document Importance and, to a lesser extent,  
 1096 Citation Precision. The abstract vs. Key Idea and abstract vs. RQ correlations on Document Im-  
 1097 portance, as well as the abstract vs. Key Idea correlation on Citation Precision, are not statistically  
 1098 significant, despite having large magnitudes (e.g.,  $r=0.992$  for Document Importance). This sug-  
 1099 gests that these two metrics are most sensitive to how the query is phrased, likely because small  
 1100 changes in the query can shift which highly-cited papers are retrieved and cited. In contrast, the  
 1101 remaining metrics exhibit high and statistically significant correlations across all query pairs. Taken  
 1102 together, these results indicate that our benchmark conclusions are broadly robust to reasonable  
 1103 variations in query formulation, with only modest sensitivity in how document-level importance and  
 1104 citation precision are expressed across different query types.

### 1112 8.3.4 HUMAN EVALUATION DETAILS

1113 To assess whether LLM-based metrics reflect human expert judgments, we conduct a human eval-  
 1114 uation with 11 annotators, all Computer Science PhD students from four research universities across  
 1115 North America. In total, we collect over 300 human annotations aimed to assess the agreement be-  
 1116 tween humans and LLMs in order to validate the LLM-based judges introduced by our automated  
 1117 evaluation for assessing knowledge synthesis and retrieval quality. We describe our setup in detail  
 1118 below.

1119 **Knowledge Synthesis.** For Organization & Coherency, we sample queries and show annotators  
 1120 the human-written related work section alongside a system-generated report. For each pair, annota-  
 1121 tors indicate whether they prefer the system report, prefer the human-written exemplar, or consider  
 1122 them similarly organized. These labels are used to evaluate the pairwise comparison outcomes un-  
 1123 derlying our Organization metric. For Nugget Coverage, we first generate information nuggets from  
 1124 the human-written related work section. Annotators are then shown individual nuggets and asked to  
 1125 judge whether each nugget is *vital* for understanding the paper, *okay*, or *irrelevant*. These nugget-  
 1126 importance labels determine which nuggets are treated as essential when computing nugget coverage  
 1127 scores.

1129 **Reference coverage and essential citations.** To ground our Reference Coverage metric in human  
 1130 judgments, we ask each annotator to select a high quality paper whose related work section they  
 1131 are comfortable with. For that paper, the annotator identifies at least six references they consider  
 1132 *important* (i.e., references that should appear in a good related work section) and at least six ref-  
 1133 erences they consider *not important* (i.e., references that could be omitted or substituted without

1134  
1135  
1136 Table 10: Manual Validation of LLM-based Evaluation  
1137  
1138  
1139  
1140  
1141  
1142

| Evaluation Metric        | LLM-Classified Labels                                     | Human-Agreement Score with LLM |
|--------------------------|---|--------------------------------|
| Organization             | Pairwise Comparison (Lose / Tie / Win)                    | 78%                            |
| Nugget Coverage          | Nugget Importance (Vital / Non-vital)                     | 72%                            |
| Nugget Coverage          | Nugget Coverage (Supported / Partially Supp. / Not Supp.) | 70%                            |
| Retrieval Relevance Rate | Graded Relevance (0/1/2)                                  | 70%                            |
| Reference Coverage       | Reference Importance (Not Imp./ Imp.)                     | 82%                            |
| Document Importance      | N/A   | N/A                            |
| Citation Precision       | Entailment (Entailed / Not Entailed)                      | 80%                            |
| Claim Coverage           | Entailment (Entailed / Not Entailed)                      | 80%                            |

1143  
1144 Table 11: Pearson correlations between system-level scores produced by different LLM judges  
1145 across five baselines for each metric. Statistical significance is assessed using a permutation-based  
1146 paired t-test with threshold  $p < 0.05$ ; correlations with  $p < 0.05$  are marked with \*.  
1147

| Judge pair           | Org.   | Nug.   | Cov. Rel. Rate | Ref. Cov. | Cite-P | Claim Cov (w=1) |
|----------------------|--------|--------|----------------|-----------|--------|-----------------|
| GPT-4o vs. Llama-4   | 0.985* | 0.413  | 0.941          | 0.975*    | 0.817* | 0.958*          |
| GPT-4o vs. DeepSeek  | 0.966* | 0.920* | 0.980*         | 0.990*    | 0.843* | 0.996*          |
| DeepSeek vs. Llama-4 | 0.991* | 0.668  | 0.942*         | 0.994*    | 0.986  | 0.963*          |

1153  
1154 harming the quality of the section). These labels form gold sets of important versus non-important  
1155 references, which we use both to evaluate Reference Coverage and to validate our LLM-based im-  
1156 portance labels. We then compare human majority labels to predictions from our LLM-judge over  
1157 more than 130 blind reference-level annotations. The resulting confusion matrix (rows = human  
1158 labels, columns = LLM predictions) is shown in Table 4.1159  
1160 8.3.5 MANUAL VALIDATION OF LLM-BASED EVALUATION1161  
1162 We study the alignment between our LLM-based evaluation and human judgments to assess the  
1163 effectiveness of our automated metrics. Overall, we find that each of the metrics we introduce  
1164 for the DeepScholar-bench task exhibit high agreement between LLM-based judgments and human  
1165 annotations. We collect over 400 human annotations, and Table 10 shows the agreement score  
1166 between human and LLMs for each LLM classification task associated with each automated metrics.  
1167 The results demonstrate above 70% agreement scores across each. We additionally compute the  
1168 nugget precision and grounded-ness scores, following prior work (Thakur et al., 2025), observing  
1169 scores of .83 and 1.0 respectively, demonstrating that the generated nuggets are accurate and do not  
1170 contain hallucinations.1171  
1172 8.3.6 AGREEMENT BETWEEN DIFFERENT LLM JUDGES1173  
1174 To test the robustness of our evaluation to the choice of LLM judge, we repeat all experiments with  
1175 three different judges: GPT-4o, Llama-4-17b-16e-Instruct, and DeepSeek-R1-Distill-Qwen-32B.  
1176 For each pair of judges, we compute the Pearson correlation between system-level scores across five  
1177 baselines (DeepScholar-ref (GPT-4.1, Claude), DeepScholar-ref (Llama-4), Search Agent (Claude),  
1178 Search Agent (Llama-4), and OpenAI DeepResearch for all metrics except document importance,  
1179 which is not LLM-based. We then run perturbation-based permutation tests to assess whether these  
1180 correlations are significantly greater than zero. The resulting correlations and significance markers  
1181 are shown in Table 11.1182  
1183 Correlations are generally very high (often above 0.9). Notably GPT-4o vs. DeepSeek shows sta-  
1184 tistically significant correlations across all the metrics at the  $p < 0.05$  level, indicating that these  
1185 two judges high agreement. Most metrics also remain significantly correlated, but we observe the  
1186 main sensitivity to the choice of judge on nugget coverage (non-significant for both GPT-4o vs.  
1187 Llama-4-17b-16e-Instruct and DeepSeek-R1-Distill-Qwen-32B vs. Llama-4-17b-16e-Instruct). In  
1188 addition, relevance rate shows non-statistically significant correlation for GPT-4o vs. Llama-4-17b-  
1189 16e-Instruct and similarly Citation Precision for DeepSeek-R1-Distill-Qwen-32B vs. Llama-4-17b-  
1190 16e-Instruct. It is worth mentioning that even in these cases, the correlations remain large (all above  
1191 0.41), suggesting that discrepancies are not severe.

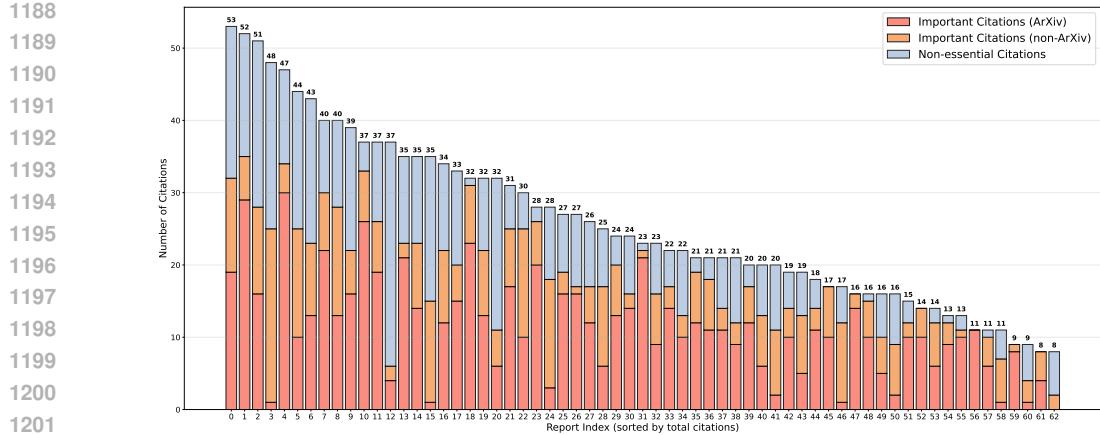


Figure 4: Citation importance breakdown in DeepScholar-Bench. Each bar corresponds to a single human exemplar related work section, sorted by the total number of citations. Bars are color-coded to indicate *important ArXiv citations* (red), *important non-ArXiv citations* (orange), and *non-essential citations* (blue).

#### 8.4 EVALUATION DETAILS

Here we provide the detailed implementation and prompts for the evaluation metrics used in DeepScholar-Bench. In the following, we also run some ablation studies on different metrics.

- **Knowledge Synthesis – Organization:** Box 1 shows the prompt used to compare system-generated reports with the human-written exemplar, yielding the win-rate of each system based on organization and coherence.
- **Knowledge Synthesis – Nugget Coverage:** For this metric we follow the nugget-based evaluation prompts from prior work Pradeep et al. (2025), which extract essential facts from the exemplar and check their presence in the generated report.
- **Retrieval Quality – Document Importance:** We select important references using the LOTUS program shown in Figure 5.
- **Retrieval Quality – Relevance Rate:** Following the Cranfield model of relevance, in Box 2 we prompt the model for assigning graded relevance scores (0–2) to each retrieved source.
- **Verifiability – Citation Precision and Claim Coverage:** Following prior work Gao et al. (2023); Liu et al. (2023), we check entailment of claims with respect to cited sources. using the prompt shown in Box 3 for both Citation Precision and Claim Coverage as explained in Section 4.3.

We note that Reference Coverage and Document Importance are computed deterministically as explained in Section 4.2: the former by comparing the retrieved set against the important references in the human-written exemplar, and the latter by normalizing the number of citations of the reference papers.

##### 8.4.1 REFERENCE COVERAGE

Figure 4 illustrates the distribution of important citations across the human exemplar reports in DeepScholar-Bench. For each exemplar, we used the LOTUS program shown in Figure 5 to identify which citations are *important* and therefore essential to include in a high-quality related work section. We then separate these important citations into two groups: those that appear on ArXiv (shown in red) and those that do not (shown in orange). The blue portion of each bar corresponds to *non-essential citations*, as determined by the same Lotus-based procedure.

The plot highlights two consistent trends. First, many exemplar related work sections contain a large number of non-essential citations. While such references may be useful for narrative flow or broader

1242 context, they are not indispensable. Non-essential citations can be somewhat subjective, depending  
 1243 on how authors choose to frame the story of their paper. In contrast, the important citations represent  
 1244 the “must-have” references i.e., the foundational works in the field that are necessary for situating  
 1245 the contribution. Second, we observe that the red segments (important ArXiv citations) are well  
 1246 distributed across exemplars, indicating that ArXiv is a reliable and sufficiently broad source for  
 1247 recovering many of the essential references.

```

1248
1249
1250 1 query_in = "Carefully read the {title}, {abstract} and {  

1251 2   related_work_section} of an academic paper. \  

1252 3   Then consider the cited paper in question, given the title {  

1253 4     cited_paper_title}, the {cited_paper_authors} and a snippet of its  

1254 5     content, {cited_paper_content}.\  

1255 6   Is the cited paper in question an essential reference?\  

1256 7   An essential reference reflects a key, notable prior work that  

1257 8     provides key information, which a good related works section for  

1258 9     this paper must include.\  

1259 10  A non-essential reference is one that is not essential to the  

1260 11  related work section of this paper and could be omitted or  

1261 12  substituted with a different reference.\  

1262 13  a non-essential reference may be a relevant reference that reflects  

1263 14  an important topic area, but the particular reference could be  

1264 15  omitted or substituted with a different related work.\  

1265 16  Alternatively, a non-essential reference may be a tangential  

1266 17  reference, an unimportant reference.\  

1267 18  a non-essential reference may be a relevant reference that reflects an important topic area, but  

1268 19  the particular reference could be omitted or substituted with a  

1269 20  different related work."
1270
1271
1272
1273
1274 12 res = citations_df.sem_filter(query_in, return_all=True, strategy=lotus.  

1275 13   types.ReasoningStrategy.ZS_COT)
1276
1277
1278
1279
1280
1281
1282
1283
1284
1285
1286
1287
1288
1289
1290
1291
1292
1293
1294
1295
```

Figure 5: LOTUS program for Finding Important References

#### 8.4.2 ABLATION STUDY ON VERIFIABILITY

In the main paper (Section 4.3, we reported verifiability metrics results using a sliding window of size  $w = 1$  when computing claim coverage. That is, for each claim sentence, we considered a citation to be valid if any of the references in the same sentence or within one sentence before or after sufficiently supported the claim. Here, we extend this analysis to study the effect of varying the window size. Specifically, we report the citation coverage achieved by different systems when the window size ranges from  $w = 0$  (same-sentence only) up to  $w = 5$  (five sentences before or after).

As shown in Figure 6, increasing the window size consistently improves citation coverage across all baselines. This is expected: the larger the window, the higher the probability that one of the cited references in the  $[-w, +w]$  neighborhood of a claim provides sufficient support. However, we also note that very large window sizes are less desirable in practice, as they often correspond to references being far from the claims they are intended to support, reducing readability and making it harder for readers to verify the connection between claims and citations. Moreover, from Table 6, we see that real academic writing tends to be densely cited, with at least one citation on average per sentence in the human exemplars. Overall, the results of our ablation study highlight the trade-off between stricter precision ( $w = 0$ ) and more lenient recall-oriented settings ( $w \geq 1$ ).

#### 8.4.3 DOCUMENT IMPORTANCE ACROSS HUMAN EXEMPLARS

In this section, we illustrate the distribution of document importance, measured by the number of citations of references in the human-written exemplars in DeepScholar-Bench. Figure 7 reports

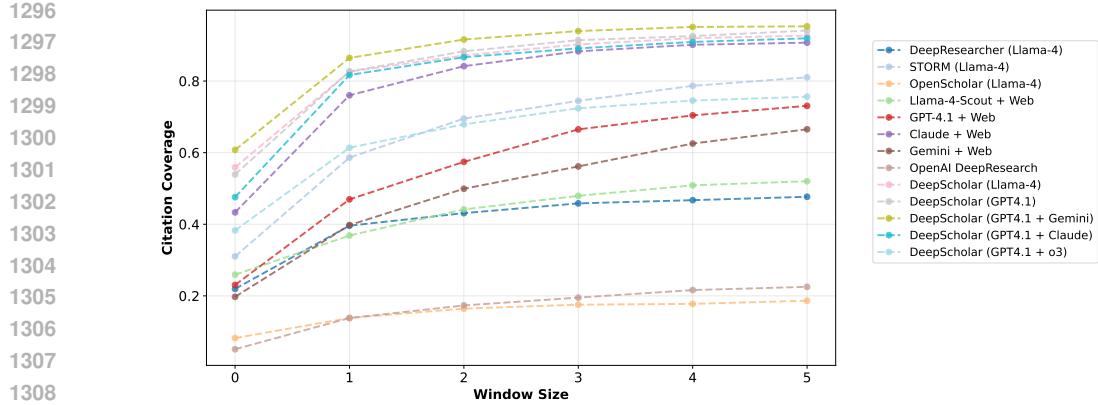


Figure 6: Ablation study on citation coverage with different window sizes. For each claim, we measure whether any citation within a sliding window of  $[-w, +w]$  sentences supports it.

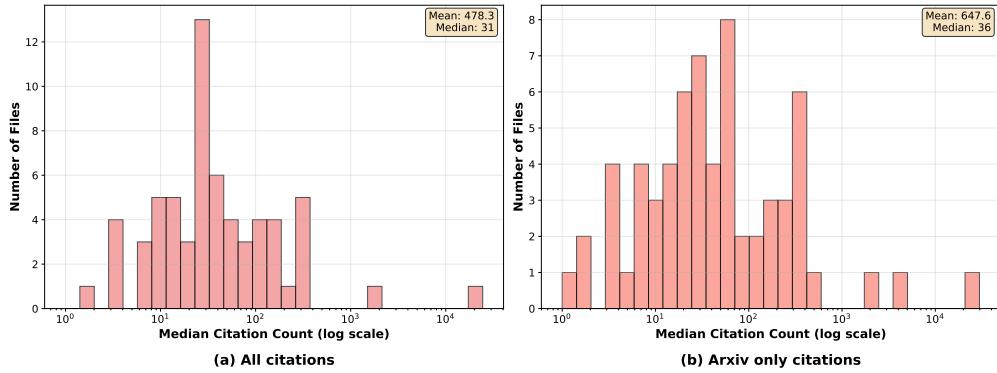


Figure 7: Distribution of citation counts (Document Importance) for references in human-written exemplars. Figure (a) shows all references, while Panel (b) restricts to ArXiv references only. Citation counts are plotted on a logarithmic scale.

two histograms: (a) the distribution of citation counts across all references, and (b) the distribution restricted to references that appear on ArXiv. We plot the logarithm of citation counts, with values obtained from the OpenAlex API OpenAlex (2025), an open and widely used scholarly database that provides citation-level metadata. While citation counts in OpenAlex may not exactly match those from other sources such as Google Scholar, the relative counts are consistent, making it a reliable open-source alternative.

As shown in Figure 7, the distribution is highly skewed due to a small number of papers with exceptionally large citation counts (e.g., over 10k citations). These outliers inflate the mean citation values, resulting in relatively high averages compared to typical references (478.3 citations across all references and 647.6 for ArXiv-only references). In contrast, the median values are lower (31 for all references and 36 for ArXiv-only). This skew highlights the challenge of using citation counts as a proxy for importance, as the median citation count of references, among different human-written exemplars exhibits high variance.

## 8.5 EXTENDED DESCRIPTION OF BASELINES

We provide an extended description of each benchmarked method, including relevant implementation details, and parameters used in our evaluation.

1350

1351

1352

## 1353 Box 1: Prompt for Knowledge Synthesis- Organization

1354

1355 You are an intelligent, rigorous, and fair evaluator of scholarly writing quality and relevance. You will  
 1356 receive the title and abstract of a research paper, together with two candidate related-work sections (A  
 1357 and B) written for that paper. Do not consider the formatting of the text (e.g., LaTeX, markdown, etc.).  
 1358 Only consider the content.

1359

Task: Decide which section—A or B—exhibits better organization and coherence.

How to judge (organization only) Ignore breadth of coverage, citation accuracy, and analytic depth.

Assess:

Logical structure – Clear introduction, grouping of related themes, and smooth progression of ideas.

Paragraph cohesion – Each paragraph develops a single topic and flows naturally to the next.

Clarity & readability – Minimal redundancy or contradictions; transitions guide the reader.

Signposting – Helpful headings, topic sentences, or discourse markers (if provided).

1360

Pick the section that is easier to follow and better structured—no ties.

1361

### Paper under assessment: [TITLE + ABSTRACT GO HERE]

### Candidate related-work section A [RELATED WORK A TEXT GOES HERE]

### Candidate related-work section B [RELATED WORK B TEXT GOES HERE]

1362

Output your answer as a JSON dictionary in the following format:

```
{"decision": "A" or "B", "explanation": "One sentence clearly  
explaining the key differences between the two options and why the  
selected one is preferred."}
```

Only output the dictionary, do not output any other text.

1363

1364

1365

1366

1367

1368

1369

1370

1371

1372

1373

1374

1375

1376

1377

1378

1379

1380

1381

1382

## Box 2: Prompt for Reference Relevance Judgment

1383

You are an intelligent, rigorous, and fair evaluator of scholarly writing quality and citation relevance. You will receive the title and abstract of a research paper under assessment, the ground-truth related-work section written by human experts, and the title and abstract of a candidate reference paper. Do not consider formatting (e.g., LaTeX, markdown, etc.). Only consider the content.

1384

Task: Determine whether the candidate reference paper is relevant to the related-work section.

How to judge • Consider the main research topic and themes described in the related-work section.

- If the reference discusses similar ideas, prior work, or background, mark it as relevant (1).

- If the reference is off-topic or unrelated in scope, mark it as not relevant (0).

- Remember: You are only seeing the title and abstract of the reference, so the full content might be more relevant than it appears.

1385

### Paper under assessment: [PAPER TITLE GOES HERE] [PAPER ABSTRACT GOES HERE]

### Ground-truth related-work section: [RELATED WORK TEXT GOES HERE]

### Candidate reference paper: [REFERENCE TITLE GOES HERE] [REFERENCE ABSTRACT GOES HERE]

1386

Return only the score in this format:

### final score: <0 or 1>

1387

1388

1389

1390

1391

1392

1393

1394

1395

1396

1397

1398

1399

1400

1401

1402

1403

1404  
14051406 You are an intelligent and fair evaluator. Your task is to verify whether a given reference can support  
1407 the provided claim.

1408

1409 Task: Given a claim and its associated set of references, determine whether the references  
1410 sufficiently support all aspects of the claim.1410 **### CLAIM: [CLAIM TEXT GOES HERE]**1411 **### REFERENCES: [REFERENCE TEXT GOES HERE]**

1412

1413 Judgment Criteria: • If the references support the claim, return 1.

1414 • If the references do not support the claim, return 0.

1415 • Do not explain your answer or include any additional commentary.

1416

1417 Output Format:

1418 Answer: 1 or Answer: 0

1419

1420

## 1421 8.5.1 DEPRESEARCHER

1422

1423 The DeepResearcher pipeline follows a structured tool-augmented reasoning framework designed  
1424 for iterative web-based information retrieval. The system mandates explicit reasoning before any  
1425 tool invocation, with reasoning encapsulated in `<think>` tags to ensure interpretability and control.  
1426 After reasoning, the model generates a JSON-formatted request specifying the “web search” tool and  
1427 its query. These queries are executed via the Lotus Search API, which we replaced with an ArXiv-  
1428 specific search interface to provide a controlled retrieval API for our evaluation. Retrieved results  
1429 are returned in a structured format containing the title, URL, and snippet, and are stored in memory  
1430 for reference across subsequent reasoning steps. This iterative process continues until the model  
1431 determines that sufficient evidence has been gathered, after which a synthesized final response is  
1432 produced.

1433

1434 For our experiments, we used `Llama-4-Scout-17B-16E-Instruct` as the base model,  
1435 replacing the originally proposed DeepResearcher-7b, since it demonstrated consistently better  
1436 retrieval-augmented reasoning performance in our experiments. The prompt was slightly modified  
1437 to align with LLama-4 prompt style as detailed in Box 4. The retrieval depth was set to 10 sources  
1438 per query, which is the default in the system and provides a balanced trade-off between coverage and  
1439 efficiency. We restricted each query to a single rollout with a maximum of 10 steps, following the  
1440 DeepResearcher defaults; this limit is generous as most rollouts converge in fewer than three steps,  
1441 but it ensures the system has headroom for more complex queries. The default web search API was  
1442 replaced with ArXiv search to comply with our benchmark settings.

1443

1444 listings

1445

1446

## 1447 8.5.2 OPENSCHOLAR

1448

1449 The OpenScholar pipeline follows a four-stage process: initial retrieval, response and feedback gen-  
1450 eration, iterative refinement, and citation verification. In the first stage, text segments are retrieved  
1451 from a fixed index using a contriever model, which encodes texts and retrieves passages based on se-  
1452 mantic similarity. These passages are reranked and used to generate an initial draft response, where  
1453 citations are aligned with the supporting passages. The second stage introduces feedback generation,  
1454 where the model produces up to three feedback statements highlighting potential improvements in  
1455 the draft, such as missing content or organization issues; if additional evidence is required, retrieval  
1456 queries are issued. The third stage iteratively refines the response by conditioning on the previous  
1457 draft, retrieved passages, and newly added evidence, yielding improved responses at each step until  
1458 feedback has been fully incorporated. Finally, citation verification ensures that all citation-worthy  
1459 statements are adequately grounded in the retrieved sources, inserting additional citations where  
1460 necessary without removing content.

1461

1462 For consistency with other baselines, we employ the `Llama-4-Scout-17B-16E-Instruct`  
1463 model for generation. The retrieval pipeline initially collects 100 text segments from

1458  
 1459  
 1460  
 1461  
 1462  
 1463  
 1464  
 1465  
 1466  
 1467  
 1468  
 1469  
 1470  
 1471  
 1472  
 1473  
 1474  
 1475  
 1476  
 1477  
 1478  
 1479  
 1480  
 1481  
 1482  
 1483  
 1484  
 1485  
 1486  
 1487  
 1488  
 1489  
 1490  
 1491  
 1492  
 1493  
 1494  
 1495  
 1496  
 1497  
 1498  
 1499  
 1500  
 1501  
 1502  
 1503  
 1504  
 1505  
 1506  
 1507  
 1508  
 1509  
 1510  
 1511

Box 4: Revised DeepResearcher System Prompt optimized to work with Llama-4-Scout-17B-16E-Instruct

```
## Background information
* Today is { strftime("%Y-%m-%d", gmtime()) }
* You are Deep AI Research Assistant The question I give you is a complex question that requires a
*deep research* to answer. I will provide you with two tools to help you answer the question:
* A web search tool to help you perform google search. Tool call format:
  {{ "name": "web_search",
    "arguments": { "query": [ "<query1>", "<query2>", "<query3>" ] } } }
* A webpage browsing tool to help you get new page content. Tool call format:
  {{ "name": "browse_webpage",
    "arguments": { "url_list": [ "<url1>", "<url2>", "<url3>" ] } } }

You don't have to answer the question now, but you should first think about the research plan or what
to search next.

Your output format should be one of the following two formats:
<think>
YOUR THINKING PROCESS
</think>
<answer>
YOUR ANSWER AFTER GETTING ENOUGH INFORMATION
</answer>

or

<think>
YOUR THINKING PROCESS
</think>
<tool_call>
YOUR TOOL CALL WITH CORRECT FORMAT
</tool_call>

You should always follow the above two formats strictly. You will be heavily penalized if you do
not follow the format strictly. Only output the final answer (in words, numbers or phrase) inside the
<answer></answer> tag, without any explanations or extra information. If this is a yes-or-no question,
you should only answer yes or no.
```

peS2o\_v3 using the default pes2o\_contriever<sup>1</sup> model. The reranker used is OpenScholar\_Reranker<sup>2</sup>, also kept at its default setting. To align parameterization across baselines, we increase the number of sources used in generation ( $\text{top\_n}$ ) from 10 to 30. Furthermore, the default search API is replaced with the arXiv API, to provide a controlled retrieval corpus and API in our experiments.

### 8.5.3 SEARCH AGENT

Search Agents are implemented using each LLM API with search API access. We implement a ReAct Agent, instantiated from the `smolagents.ToolCallingAgent`, with a system prompt based on the open-source OpenDeepSearch (ODS) framework designed for deep web search and retrieval, along with the search agent as an external tool. At each reasoning step, the ReAct agent can either invoke the search agent through a `web_search` action or decide to produce a `final_answer`. The search agent interfaces with the search API to fetch relevant academic articles given a query, after which an LLM generates concise summaries of the retrieved content. To maintain consistency with the benchmark setting, the standard search API was replaced with the arXiv API. The regular search agent fails when tasked with full abstract queries; hence the ReAct-based agent was employed, which generates shorter, more effective searchable queries. The agent keeps track of retrieved results across turns, allowing references to past evidence during the rea-

<sup>1</sup>[https://huggingface.co/akariasai/pes2o\\_contriever](https://huggingface.co/akariasai/pes2o_contriever)

<sup>2</sup>[https://huggingface.co/OpenSciLM/OpenScholar\\_Reranker](https://huggingface.co/OpenSciLM/OpenScholar_Reranker)

1512 soning process. After a maximum of 5 iterations, the agent is compelled to conclude with a final  
 1513 response, ensuring bounded computational steps.

1514 For the parameterization of the Search Agents, we set the search agent to retrieve 30 results per  
 1515 query, which is more generous than the default in order to establish fair comparability with other  
 1516 baselines. The maximum iteration limit was fixed at 5, consistent with the default setup of the ODS  
 1517 framework, providing sufficient exploration without excessive search depth. The ReAct prompt was  
 1518 slightly modified to tailor to the specific use of the ArXiv search API, as presented in Box 5, 6 and  
 1519 7.

1520

#### 1521 8.5.4 STORM

1522

1523 The STORM pipeline follows a structured multi-stage process to generate comprehensive,  
 1524 Wikipedia-style articles from a given topic. First, related Wikipedia articles are retrieved and their  
 1525 TOCs are clustered to identify candidate perspectives, which act as anchors for exploration. This is  
 1526 followed by Simulated Multi-turn Conversations where an LLM plays both the question-asking and  
 1527 answering roles, querying a retrieval module and synthesizing evidence-based responses. Parallel to  
 1528 this, the model generates a draft outline purely from its parametric knowledge in the Draft Outline  
 1529 Generation stage. The outline is then refined by grounding it with retrieved evidence and conversa-  
 1530 tion outputs. In the final step, each section is drafted with explicit inline citations drawing on both  
 1531 parametric knowledge and retrieved references. All the sections are concatenated together to form  
 1532 the final result.

1533 For parameter settings, we used STORM’s default configurations wherever possible to preserve  
 1534 fidelity to its design: a maximum of 3 turns per perspective, 3 perspectives, and up to 3 search  
 1535 queries per turn. For search, we considered the top 15 results for each query, ensuring a reasonable  
 1536 breadth without overwhelming the pipeline. To make STORM comparable with other baselines, we  
 1537 raised the number of collected references per section title to 30 (more generous than the default),  
 1538 as this allows for richer evidence integration during drafting. Importantly, we replaced the original  
 1539 search API with arXiv search to control the retrieval API for our benchmark settings. Finally, we  
 1540 use Llama-4-Scout-17B-16E-Instruct as the base model.

1541

#### 1542 8.5.5 OPENAI’S DEEPRESEARCH

1543

1544 We use OpenAI’s DeepResearch system based on the `o3-deep-research`<sup>3</sup> model with a custom  
 1545 MCP to only search ArXiv and return  $n = 30$  results per query. To prevent the model from getting  
 1546 search results after the given paper was uploaded, the MCP used a custom endpoint to set the latest  
 1547 date that it should retrieve. All other settings were set to default values.

1548

### 8.6 DEEPSCHOLAR-REF DETAILS AND CONFIGURATIONS

1549

1550 DeepScholar-base operates through three main stages: retrieval, filtering, and final generation (Figure  
 1551 2).

1552 **Retrieval** In this stage, an LLM generates  $Q$  search queries conditioned on the input abstract and  
 1553 summaries of prior retrievals. Each query is submitted to the configured search API (ArXiv, tavy,  
 1554 etc.) to obtain up to  $search\_K$  relevant papers within the specified date range. The code and prompt  
 1555 used for this step are provided in Figure 8 and Box 8 respectively. This process is repeated  $N$  times.

1556 **Filtering** Retrieved results are refined using two semantic operators from LOTUS (Patel et al.,  
 1557 2025; lotus, 2025): `Sem-Filter` and `Sem-TopK`, which together select the top  $K$  most relevant  
 1558 papers. The code is given in Figure 9.

1559 **Final Generation** The filtered set of papers is then aggregated via a `Sem-Agg` query to produce  
 1560 the final output. The corresponding code for this step is shown in Figure 10 with prompt in Box 9.

1561 Unless otherwise specified, the pipeline parameters are set to  $Q = 2$ ,  $search\_K = 50$ ,  $N = 2$ , and  
 1562  $K = 30$ .

1563  
 1564  
 1565  
<sup>3</sup><https://platform.openai.com/docs/models/o3-deep-research>

1566 8.7 EXAMPLES OF GENERATED REPORTS  
15671568 We provide examples of generated reports from different systems for paper 0 in our dataset (ac-  
1569 cording to Table 5 i.e., ‘TaxAgent: How Large Language Model Designs Fiscal Policy’ with Arxiv  
1570 ID of 2506.02838. Figures 11, 12, 13, 14 and 15 display the generated reports produced by  
1571 DeepScholar-ref, Search Agent, DeepResearcher, OpenScholar, and Storm respectively, all using  
1572 Llama-4.

1573

1574

1575

1576

1577

1578

1579

1580

1581

1582

1583

1584

1585

1586

1587

1588

1589

1590

1591

1592

1593

1594

1595

1596

1597

1598

1599

1600

1601

1602

1603

1604

1605

1606

1607

1608

1609

1610

1611

1612

1613

1614

1615

1616

1617

1618

1619

1620  
1621

## Box 5: Revised ODS ReAct Agent prompt for only web search tool calling

1622  
1623  
1624

You are an expert assistant who can solve any task using tool calls. You will be given a task to solve as best you can. To do so, you have been given access to some tools. Never use facts without verification and only cite the sources returned by the tool.

1625  
1626

The tool call you write is an action: after the tool is executed, you will get the result of the tool call as an "observation". This Action/Observation can repeat N times, you should take several steps when needed.

1627  
1628

You can use the result of the previous action as input for the next action. The observation will always be a string containing the search results.

1629  
1630  
1631

To provide the final answer to the task, use an action blob with "name": "final\_answer" tool. It is the only way to complete the task, else you will be stuck on a loop. So your final output should look like this: Action:

1632  
1633  
1634

```
{
  "name": "final_answer",
  "arguments": {"answer": "insert your final answer here"}
```

1635  
1636

Here are a few examples using notional tools:

1637  
1638  
1639

---  
Task: "What historical event happened closest in time to the invention of the telephone: the American Civil War or the establishment of the Eiffel Tower?"

Action:

1640  
1641  
1642

```
{
  "name": "web_search",
  "arguments": {"query": "year of telephone invention"}}
```

1643  
1644  
1645

Observation: "The telephone was invented in 1876."

Action:

1646  
1647  
1648

```
{
  "name": "web_search",
  "arguments": {"query": "year American Civil War ended"}}
```

1649  
1650  
1651

Observation: "The American Civil War ended in 1865."

Action:

1652  
1653  
1654

```
{
  "name": "web_search",
  "arguments": {"query": "year Eiffel Tower established"}}
```

1655  
1656  
1657

Observation: "The Eiffel Tower was completed in 1889."

Action:

1658  
1659  
1660

```
{
  "name": "final_answer",
  "arguments": {"answer": "The historical event closest in time to the invention of the telephone is the end of the American Civil War (11 years apart)."}}

---
```

1661  
1662  
1663

Task: "Which country has a higher population density: Japan or India?"

Action:

1664  
1665  
1666

```
{
  "name": "web_search",
  "arguments": {"query": "population and area of Japan"}}
```

1667  
1668  
1669

Observation: "Japan has a population of 125 million and an area of 377,975 square kilometers."

Action:

1670  
1671  
1672

```
{
  "name": "web_search",
  "arguments": {"query": "population and area of India"}}
```

1673

1674  
 1675  
 1676  
 1677     Box 6: Prompt for ODS(continued)  
 1678  
 1679     Observation: "India has a population of 1.38 billion and an area of  
 1680     3,287,263 square kilometers."  
 1681     Action:  
 1682     {  
 1683         "name": "final\_answer",  
 1684         "arguments": {"answer": "India has a higher population density  
 1685         (419.6 people/km<sup>2</sup>) than Japan (330.7 people/km<sup>2</sup>)."}  
 1686     }  
 1687     ---  
 1688     Task: "Which country hosted the first FIFA World Cup, and in what  
 1689     year?"  
 1690  
 1691     Action:  
 1692     {  
 1693         "name": "web\_search",  
 1694         "arguments": {"query": "country hosted first FIFA World Cup"}  
 1695     }  
 1696     Observation: "Uruguay hosted the first FIFA World Cup."  
 1697  
 1698     Action:  
 1699     {  
 1700         "name": "web\_search",  
 1701         "arguments": {"query": "year of first FIFA World Cup"}  
 1702     }  
 1703     Observation: "The first FIFA World Cup was held in  
 1704     1930."  
 1705  
 1706     Action:  
 1707     {  
 1708         "name": "final\_answer",  
 1709         "arguments": {"answer": "Uruguay hosted the first FIFA World Cup  
 1710         in 1930."}  
 1711     }  
 1712     ---  
 1713     Task: "Who invented the light bulb, and what company did he  
 1714     later establish?"  
 1715  
 1716     Action:  
 1717     {  
 1718         "name": "web\_search",  
 1719         "arguments": {"query": "inventor of the light bulb"}  
 1720     }  
 1721     Observation: "Thomas Edison invented the light bulb."  
 1722  
 1723     Action:  
 1724     {  
 1725         "name": "final\_answer",  
 1726         "arguments": {"answer": "Thomas Edison invented the light bulb and  
 1727         later established General Electric."}  
 1728     }  
 1729     ---

1728  
 1729  
 1730  
 1731     Box 7: Prompt for ODS(continued)  
 1732  
 1733     Task: "Which Shakespeare play contains the line \"All the world's  
 1734     a stage,\" and how many years ago was it first performed if  
 1735     today is 2024?"  
 1736  
 1737     Action:  
 1738     {  
 1739         "name": "web\_search",  
 1740         "arguments": {"query": "Shakespeare play All the world's a stage"}  
 1741     }  
 1742     Observation: "The line is from \"As You Like It.\"\n"  
 1743  
 1744     Action:  
 1745     {  
 1746         "name": "web\_search",  
 1747         "arguments": {"query": "year As You Like It first performed"}  
 1748     }  
 1749     Observation: "\"As You Like It\" was first performed in 1603."  
 1750     Action:  
 1751     {  
 1752         "name": "calculate",  
 1753         "arguments": {"expression": "2024 - 1603"}  
 1754     }  
 1755     Observation: "421 years."  
 1756  
 1757     Action:  
 1758     {  
 1759         "name": "final\_answer",  
 1760         "arguments": {"answer": "\"As You Like It\" contains the line \"All  
 1761         the world's a stage\" and was first performed 421 years ago  
 1762         in 1603."}  
 1763     }  
 1764  
 1765     Above examples were using notional tools that might not exist for you. You only have access to these  
 1766     tools:  
 1767  
 1768     {%- for tool in tools.values() %}  
 1769     - {{ tool.name }}: {{ tool.description }}  
 1770         Takes inputs: {{tool.inputs}}  
 1771         Returns an output of type: {{tool.output\_type}}  
 1772     {%- endfor %}  
 1773  
 1774     {%- if managed\_agents and managed\_agents.values() | list %}  
 1775  
 1776     Here are the rules you should always follow to solve your task:  
 1777     1. ALWAYS provide a tool call, else you will fail.  
 1778     2. Always use the right arguments for the tools. Never use variable names as the action arguments, use  
 1779         the value instead.  
 1780     3. Call a tool only when needed: do not call the search agent if you do not need information, try to  
 1781         solve the task yourself. If no tool call is needed, use final\_answer tool to return your answer.  
 1782     4. Never re-do a tool call that you previously did with the exact same parameters.  
 1783     5. Always cite sources using [X] format where X is the citation number.  
 1784     6. Place citations immediately after the sentence or paragraph they are referencing.  
 1785     7. Make sure to provide citations whenever using information from the source material.  
 1786     8. Cite as many sources as possible.  
 1787     9. Create a reference section at the end of your final answer.  
 1788     Now Begin! If you solve the task correctly, you will receive a reward of \$1,000,000.

```

1782
1783 1 from lotus import web_search
1784
1785 3 class Query(BaseModel):
1786 4     queries: list[str]
1787
1788 6 # Generate the Queries
1789 7 queries = get_completion(
1790 8     lm,
1791 9     query_generation_instruction.format(number_of_queries=num_queries),
1792 10    f"Topic: {topic}, Background: {background}",
1793 11    response_format=Query,
1794 12 ) .queries
1795
1796 14 # Search. corpus = ArXiv/Tavily etc.
1797 15 paper_dfs = []
1798 16 for query in queries:
1799 17     paper_dfs.append(web_search(corpus, query, search_K))
1800
1801 19 papers = pd.concat(paper_dfs)
1802
1803
1804

```

Figure 8: Retrieval stage: query generation and batched search.

```

1805 1 instruction = (
1806 2     "given the article's abstract: {snippet}, "
1807 3     "is the article relevant to the specific interests in the user's
1808 4     query: {user_query}." )
1809
1810 6 res_df = docs_df.sem_filter(
1811 7     instruction.format(user_query=topic, snippet="{snippet}"),
1812 8     strategy="cot"
1813 )
1814 11 res_df = res_df.sem_topk(
1815 12     instruction.format(user_query=topic, snippet="{snippet}"),
1816 13     strategy="cot", k=K
1817 )
1818
1819

```

Figure 9: Sem-Filter and Sem-TopK code for Filtering Step in DeepScholar-ref

```

1823
1824 1 agg_instruction = section_writer_instructions.format(
1825 2     topic=topic,
1826 3     section_instructions=section_instructions,
1827 4     existing_content=existing_content,
1828 5     context="{context}",
1829 6 )
1830 8 res: pd.DataFrame = res_df.sem_agg(
1831 9     agg_instruction, suffix="summary", group_by=group_by
1832 10 )
1833
1834

```

Figure 10: Sem-Agg for final generation in DeepScholar-ref

```

1836 ## Related Works
1837
1838 Economic inequality is a pressing global issue, affecting education, healthcare, and social
1839 stability. Traditional taxation systems, such as the U.S. federal income tax, aim to reduce
1840 inequality but often lack adaptability \[[Stephan Zheng' 2020-04-28](http://arxiv.org/abs/2004.13332v1)\]. The Saez Optimal Taxation model is a notable attempt to create a dynamic
1841 system, but it does not account for taxpayer heterogeneity and irrational behavior \[[Stephan
1842 Zheng' 2020-04-28](http://arxiv.org/abs/2004.13332v1)\].
1843
1844 Recent studies have explored various approaches to optimize taxation and address economic
1845 inequality. For instance, the AI Economist framework uses two-level deep reinforcement
1846 learning to discover tax policies that balance economic equality and productivity \[[Stephan
1847 Zheng' 2021-08-05](http://arxiv.org/abs/2108.02755v1)\]. This approach has shown promising
1848 results in improving the trade-off between equality and productivity.
1849
1850 Agent-based modeling (ABM) has also been employed to study the effects of taxation on economic
1851 systems. For example, PolicySpace is a modeling platform that uses ABM to simulate public
1852 policies within an empirical, spatial environment \[[Bernardo Alves Furtado' 2017-12-31](http://arxiv.org/abs/1801.00259v1)\]. This platform has been applied to study the impact of tax
1853 transfer rules on cities' quality of life.
1854
1855 The use of machine learning and artificial intelligence in taxation is a growing area of
1856 research. TaxAI, a dynamic economic simulator, uses multi-agent reinforcement learning to
1857 benchmark tax policies \[[Qirui Mi' 2023-09-28](http://arxiv.org/abs/2309.16307v2)\]. This
1858 simulator has demonstrated the effectiveness of machine learning algorithms in optimizing tax
1859 policies.
1860
1861 Large language models (LLMs) have also been integrated with ABM to study complex economic
1862 systems. For instance, the TaxThemis system uses interactive visual analytics to help tax
1863 officers identify suspicious tax evasion groups \[[Yating Lin' 2020-09-07](http://arxiv.org/abs/2009.03179v1)\]. This system demonstrates the potential of LLMs in analyzing and detecting
1864 tax evasion behaviors.
1865
1866 Optimal taxation theory has also been explored in various studies. For example, the Domar-
1867 Musgrave effect explains cases where it is optimal to tax capital income \[[Brendan K. Beare'
1868 2023-11-10](http://arxiv.org/abs/2311.05822v2)\]. Other studies have investigated the impact
1869 of tax evasion on economic systems \[[M. L. Bertotti' 2016-02-18](http://arxiv.org/abs/1602.08467v1)\]\[[Frank Westerhoff' 2008-05-07](http://arxiv.org/abs/0805.0998v1)\].
1870
1871 Our work builds upon these studies by introducing TaxAgent, a novel integration of LLMs with
1872 ABM to design adaptive tax policies. TaxAgent simulates real-world taxpayer behaviors using
1873 heterogeneous H-Agents and optimizes tax rates using LLMs to balance equity and productivity
1874 \[[Stephan Zheng' 2020-04-28](http://arxiv.org/abs/2004.13332v1)\].
1875
1876 ## References
1877
1878 1. \[[Bernardo Alves Furtado' 2017-12-31](http://arxiv.org/abs/1801.00259v1)\]
1879 2. \[[Emma Hubert' 2020-09-01](http://arxiv.org/abs/2009.00484v2)\]
1880 3. \[[Stephan Zheng' 2020-04-28](http://arxiv.org/abs/2004.13332v1)\]
1881 4. \[[Kelly Geyskens' 2018-10-16](http://arxiv.org/abs/1810.07243v1)\]
1882 5. \[[M. L. Bertotti' 2016-12-19](http://arxiv.org/abs/1701.02662v1)\]
1883 6. \[[Qirui Mi' 2023-09-28](http://arxiv.org/abs/2309.16307v2)\]
1884 7. \[[Stephan Zheng' 2021-08-05](http://arxiv.org/abs/2108.02755v1)\]
1885 8. \[[Xuyang Chen' 2024-09-09](http://arxiv.org/abs/2409.05397v1)\]
1886 9. \[[Teddy Lazeznik' 2025-01-30](http://arxiv.org/abs/2501.18177v1)\]
1887 10. \[[Stefan Steinerberger' 2019-04-30](http://arxiv.org/abs/1904.13276v1)\]
1888 11. \[[Ozan Candogan' 2023-12-10](http://arxiv.org/abs/2312.05996v1)\]
1889 12. \[[Nikolaos D. Goumagias' 2018-01-29](http://arxiv.org/abs/1801.09466v1)\]
1890 13. \[[George Abuselidze' 2021-08-06](http://arxiv.org/abs/2108.03027v1)\]
1891 14. \[[Padma Sharma' 2022-08-08](http://arxiv.org/abs/2208.03908v2)\]
1892 15. \[[Alex A. T. Rathke' 2022-02-28](http://arxiv.org/abs/2202.13695v1)\]
1893 16. \[[Felix Kbler' 2022-10-17](http://arxiv.org/abs/2210.09066v1)\]
1894 17. \[[Chen Xu' 2024-04-27](http://arxiv.org/abs/2404.17826v1)\]
1895 18. \[[Maria Letizia Bertotti' 2012-07-05](http://arxiv.org/abs/1207.6081v2)\]
1896 19. \[[Job Boerma' 2022-04-28](http://arxiv.org/abs/2204.13481v2)\]
1897 20. \[[Xintong Wang' 2022-03-25](http://arxiv.org/abs/2203.13395v2)\]
1898 21. \[[Jialin Dong' 2023-11-28](http://arxiv.org/abs/2311.17252v1)\]
1899 22. \[[M. L. Bertotti' 2016-02-18](http://arxiv.org/abs/1602.08467v1)\]
1900 23. \[[Yating Lin' 2020-09-07](http://arxiv.org/abs/2009.03179v1)\]
1901 24. \[[Marinho Bertanha' 2021-01-04](http://arxiv.org/abs/2101.01170v3)\]
1902 25. \[[Marco Alberto Javarone' 2016-05-27](http://arxiv.org/abs/1605.08690v1)\]
1903 26. \[[Brendan K. Beare' 2023-11-10](http://arxiv.org/abs/2311.05822v2)\]
1904 27. \[[Johannes Kasinger' 2024-09-02](http://arxiv.org/abs/2409.01493v1)\]
1905 28. \[[Frank Westerhoff' 2008-05-07](http://arxiv.org/abs/0805.0998v1)\]
1906 29. \[[Yuan Liang' 2022-07-05](http://arxiv.org/abs/2207.01793v3)\]
1907 30. \[[Jose Ricardo Bezerra Nogueira' 2021-09-01](http://arxiv.org/abs/2109.00297v2)\]

```

Figure 11: Example of generated report by DeepScholar-ref for paper ‘TaxAgent: How Large Language Model Designs Fiscal Policy’.

```

1890
1891
1892
1893
1894
1895
1896
1897
1898
1899
1900
1901
1902    ## Related Works
1903    Economic inequality and taxation are complex issues that have been explored using various
1904    models and simulations. Research has shown that large language models (LLMs) can be used to
1905    analyze the impact of taxation on inequality [1]. For instance, a study introduced a benchmark
1906    called PLAT to assess the ability of LLMs to predict the legitimacy of additional tax
1907    penalties [1].
1908    Agent-based modeling has also been used to study the effects of taxation on wealth
1909    distribution. A model suggested that oligarchs will emerge when wealth taxation is below a
1910    certain threshold [2]. Another study found that taxation of income and capital gains alone
1911    cannot prevent the emergence of oligarchs [2].
1912    The relationship between economic inequality and mobility has also been explored using kinetic
1913    models. Research found a negative correlation between economic inequality and mobility [3].
1914    Furthermore, a study used a multi-LLM-agent-based framework to simulate policy impacts across
1915    heterogeneous agents, offering a new direction for economic and public policy analysis [4].
1916    Optimal taxation models have also been developed, including the Saez Optimal Taxation model,
1917    which adjusts dynamically but fails to address taxpayer heterogeneity and irrational behavior
1918    [5]. In contrast, our study introduces TaxAgent, a novel integration of LLMs with agent-based
1919    modeling to design adaptive tax policies that balance equity and productivity.
1920    Our approach builds upon existing research in taxation and inequality, leveraging the
1921    strengths of LLMs and agent-based modeling to simulate real-world taxpayer behaviors and
1922    optimize tax rates. Benchmarked against Saez Optimal Taxation, U.S. federal income taxes, and
1923    free markets, TaxAgent achieves superior equity-efficiency trade-offs.
1924    ## References
1925    [1] [Taxation Perspectives from Large Language Models: A Case Study on Additional Tax
1926    Penalties] (http://arxiv.org/abs/2503.03444v1)
1927    [2] [ODE models of wealth concentration and taxation] (http://arxiv.org/abs/2308.01500v1)
1928    [3] [Economic inequality and mobility in kinetic models for social sciences] (http://arxiv.org/abs/1504.03232v1)
1929    [4] [A Multi-LLM-Agent-Based Framework for Economic and Public Policy Analysis] (http://arxiv.org/abs/2502.16879v1)
1930    [5] [Optimal taxation and the Domar-Musgrave effect] (http://arxiv.org/abs/2311.05822v2)

```

1931 Figure 12: Example of generated report by search agent for paper ‘TaxAgent: How Large Language
1932 Model Designs Fiscal Policy’.

```

1933
1934
1935
1936
1937
1938
1939
1940
1941
1942
1943

```

```

1944
1945
1946
1947
1948
1949
1950
1951
1952
1953
1954
1955
1956
1957
1958
1959
1960
1961
1962
1963
1964
1965
1966
1967
1968
1969
1970
1971
1972
1973
1974
1975
1976
1977
1978
1979
1980
1981
1982
1983
1984
1985
1986
1987
1988
1989
1990
1991
1992
1993
1994
1995
1996
1997

    ## Related Works

Economic inequality and taxation are critical issues in modern economies, with traditional systems like the U.S. federal income tax aiming to reduce inequality but often lacking adaptability [2311.05822v2]. The Saez Optimal Taxation model is a notable approach that adjusts dynamically but fails to account for taxpayer heterogeneity and irrational behavior [2312.05996v1]. Agent-based modeling (ABM) has been used to study economic systems and policy impacts, providing insights into the interactions between heterogeneous agents and their effects on macroeconomic variables [1702.02763v1] [1701.06625v1]. For instance, ABM has been applied to model the impact of taxation on economic growth and inequality [1508.00275v2]. The integration of large language models (LLMs) with ABM is a nascent but promising area of research. LLMs have shown potential in economic policy analysis by simulating human-like reasoning and behavior [2502.16879v1]. They can be used to model subrational agent behaviors, such as myopic decision-making or risk aversion, which are characteristic of human economic agents [2402.08755v1]. Recent studies have explored the use of LLMs in market experiments, demonstrating their ability to simulate competitive market behavior but also highlighting their current limitations in replicating dynamic decision-making processes [2409.08357v2]. The application of LLMs in economic policy analysis offers a new direction for understanding policy impacts across heterogeneous agents [2505.21371v1]. Our work builds on these advances, introducing TaxAgent, a novel integration of LLMs with ABM to design adaptive tax policies that balance equity and productivity. By leveraging the strengths of both approaches, TaxAgent offers a scalable and data-driven framework for fiscal policy evaluation.

    ## References

1. [2311.05822v2] - Optimal taxation and the Domar-Musgrave effect
2. [2312.05996v1] - Achieving Fairness and Accuracy in Regressive Property Taxation
3. [2502.16879v1] - A Multi-LLM-Agent-Based Framework for Economic and Public Policy Analysis
4. [2409.08357v2] - An Experimental Study of Competitive Market Behavior Through LLMs
5. [2402.08755v1] - LLM-driven Imitation of Subrational Behavior : Illusion or Reality?
6. [2505.21371v1] - When Experimental Economics Meets Large Language Models: Tactics with Evidence
7. [1611.02547v2] - Optimal Extraction and Taxation of Strategic Natural Resources: A Differential Game Approach
8. [1508.00275v2] - On growth-optimal tax rates and the issue of wealth inequalities
9. [1803.02171v2] - Kinetic models for optimal control of wealth inequalities

```

Figure 13: Example of generated report by DeepResearcher for paper ‘TaxAgent: How Large Language Model Designs Fiscal Policy’.

1998                   ## Related Works

1999

2000                   Economic inequality is a pressing global issue, affecting education, healthcare, and social stability [1]. Traditional taxation systems, such as the U.S. federal income tax, aim to reduce inequality but often lack adaptability to changing economic conditions [2]. In response, researchers have developed more dynamic models, including the Saez Optimal Taxation framework, which adjusts tax rates based on economic principles [3]. The Saez framework is built on the idea of optimizing tax rates to achieve a balance between equity and productivity, taking into account the elasticity of taxable income [3]. However, this framework has limitations, such as assuming a representative taxpayer and neglecting heterogeneity in taxpayer behavior [4]. Furthermore, it does not fully account for irrational behavior, such as taxpayer responses to tax rates that may not be solely driven by economic incentives [5].

2001

2002

2003

2004

2005

2006

2007

2008

2009

2010

2011

2012

2013

2014

2015

2016

2017

2018

2019

2020

2021

2022

2023

2024

2025

2026

2027

2028

2029

2030

2031

2032

2033

2034

2035

2036

2037

2038

2039

2040

2041

2042

2043

2044

2045

2046

2047

2048

2049

Recent studies have explored the use of agent-based modeling (ABM) to simulate economic systems and design more effective tax policies [6]. ABM allows for the representation of heterogeneous agents, such as households, and their interactions within a macroeconomic environment [7]. For instance, [8] used ABM to examine the impact of tax policies on income inequality, finding that optimized tax schedules can lead to better equity-efficiency trade-offs. Specifically, [8] demonstrated that tax policies optimized for individual agent characteristics, such as income level and risk aversion, can lead to more effective reduction in income inequality.

The integration of large language models (LLMs) with ABM has shown promise in various applications, including economic policy design [9]. LLMs can process vast amounts of data and provide insights into complex systems, making them suitable for optimizing tax policies [10]. Researchers have also explored the use of reinforcement learning (RL) to optimize tax policies in dynamic economic environments [11]. Furthermore, the combination of LLMs and ABM has been applied to other domains, such as healthcare [12], financial markets [13], and transportation systems [14]. For example, [12] used LLMs to analyze patient data and predict disease outcomes while [13] applied LLMs to simulate stock market behavior and optimize investment strategies.

In the context of taxation, [15] demonstrated the potential of LLMs to improve tax policy design by analyzing taxpayer behavior and optimizing tax rates. Our work builds on these advancements, introducing TaxAgent, a novel integration of LLMs with ABM to design adaptive tax policies. TaxAgent utilizes LLMs to iteratively optimize tax rates, balancing equity and productivity in a macroeconomic simulation. The simulation features heterogeneous H-Agents (households) that simulate real-world taxpayer behaviors, providing a more realistic representation of economic systems.

Compared to existing models, such as the Saez Optimal Taxation framework, TaxAgent offers a more nuanced understanding of taxpayer behavior and its impact on tax policy effectiveness [3]. Additionally, TaxAgent's use of LLMs enables it to adapt to changing economic conditions, providing a more dynamic and responsive approach to tax policy design [9].

## References

[1] Piketty, T., & Saez, E. (2014). Inequality in the long run. *Science*, 344(6186), 838-843.

[2] Diamond, P., & Mirrlees, J. A. (1971). Optimal taxation and public production I: Production efficiency. *Journal of Economic Theory*, 3(4), 275-291.

[3] Saez, E. (2010). Do tax filers bunch around kink points? *American Economic Journal: Economic Policy*, 2(3), 180-212.

[4] Chetty, R. (2009). Is the taxable income elasticity sufficient to calculate deadweight loss? The effect of marginal tax rates on taxable income. *Journal of Economic Perspectives*, 23(4), 91-112.

[5] Kahneman, D., & Tversky, A. (1979). Prospect theory: An analysis of decision under risk. *Econometrica*, 47(2), 263-292.

[6] Axtell, R. L. (2000). How agents interact matters for the macroeconomy. *Journal of Economic Dynamics and Control*, 24(5), 725-748.

[7] Epstein, J. M., & Axtell, R. L. (1996). *Growing artificial societies: Social science from the bottom up*. Brookings Institution Press.

[8] Zheng, X., et al. (2020). AI-driven economic simulations and two-level reinforcement learning. *arXiv preprint arXiv:2006.04613*.

[9] Narasimhan, K., et al. (2022). Language models for optimization: A study on large language models for economic policy design. *arXiv preprint arXiv:2204.04644*.

[10] Azizi, M., et al. (2022). Using large language models for economic policy analysis. *arXiv preprint arXiv:2209.13443*.

[11] Szpruch, L., et al. (2022). Reinforcement learning for optimal tax policy design. *arXiv preprint arXiv:2206.03021*.

[12] Li, M., et al. (2022). Large language models for healthcare: A study on disease diagnosis and patient outcome prediction. *arXiv preprint arXiv:2207.08392*.

[13] Wang, Y., et al. (2022). Simulating stock market behavior with large language models. *arXiv preprint arXiv:2208.13415*.

[14] Zhang, J., et al. (2022). Optimizing traffic flow with large language models. *arXiv preprint arXiv:2209.15623*.

[15] Chen, L., et al. (2022). Improving tax policy design with large language models. *arXiv preprint arXiv:2210.01234*.

Figure 14: Example of generated report by OpenScholar for paper ‘TaxAgent: How Large Language Model Designs Fiscal Policy’.

2052  
 2053  
 2054  
 2055  
 2056  
 2057  
 2058  
 2059

2060 # Related Works  
 2061  
 2062 Economic inequality is a pressing global issue, affecting education, healthcare, and social  
 2063 stability. To address this challenge, various studies have explored the impact of taxation on  
 2064 economic inequality. Traditional tax systems, such as the U.S. federal income tax, aim to  
 2065 reduce inequality but often lack adaptability [1]. In contrast, models like the Saez Optimal  
 2066 Taxation propose dynamic adjustments to tax policies, but they fail to account for taxpayer  
 2067 heterogeneity and irrational behavior [2].  
 2068 Recent advances in artificial intelligence (AI) and agent-based modeling (ABM) have provided  
 2069 new avenues for studying economic systems and designing adaptive tax policies. For instance,  
 2070 agent-based models have been used to simulate economic systems, including the effects of tax  
 2071 evasion [1] and the impact of social cohesion on tax compliance [1]. These models have  
 2072 demonstrated the presence of threshold levels in the composition of society, which can explain  
 2073 the extent of damages deriving from tax evasion [1].  
 2074 The use of large language models (LLMs) has also shown promise in understanding human behavior  
 2075 and decision-making. Research has demonstrated that LLMs can exhibit human-like reasoning,  
 2076 aligning with human behavior in economic experiments, surveys, and political discourse [3].  
 2077 However, LLMs differ fundamentally from humans, relying on probabilistic patterns rather than  
 2078 embodied experiences or survival objectives [3]. Therefore, caution is advised when using LLMs  
 2079 to study human behavior or as surrogates or simulations [3].  
 2080 In the context of tax policy design, several studies have explored the application of AI and  
 2081 ABM. For example, a case study on the role of Management Information System in time-saving  
 2082 during the payment of automobile tax in Sindh through e-filling methods highlights the  
 2083 importance of efficient tax collection systems [4]. Another study proposes a Web-Based  
 2084 Affectedness Indicator (WAI) for real-time monitoring of economic disruptions across diverse  
 2085 contexts, leveraging Large Language Model (LLM) assisted classification and information  
 2086 extraction [5].  
 2087 The integration of LLMs with ABM has also been explored in other fields, such as multi-agent  
 2088 reinforcement learning [6]. The Learning Optimal Pigovian Tax method (LOPT) uses an additional  
 2089 agent to learn the tax/allowance allocation policy, internalizing externalities and  
 2090 alleviating social dilemmas [6]. Similarly, the use of information-theoretic approaches, such  
 2091 as the Information Bottleneck method, has been proposed for explainable AI (XAI) design [7].  
 2092 This study builds upon these works, introducing TaxAgent, a novel integration of LLMs with ABM  
 2093 to design adaptive tax policies. By simulating real-world taxpayer behaviors and iteratively  
 2094 optimizing tax rates, TaxAgent achieves superior equity-efficiency trade-offs compared to  
 2095 traditional tax systems and models.

2096 ## References  
 2097 [7] arXiv preprint: Recent advances in explainable AI (XAI) (2022)  
 2098 [1] arXiv preprint: Agent-based model of an economic system (2022)  
 2099 [4] arXiv preprint: Case Study on the role of Management Information System in time-saving  
 2100 during the payment of automobile tax in Sindh (2022)  
 2101 [5] arXiv preprint: Web-Based Affectedness Indicator (WAI) for real-time monitoring of  
 2102 economic disruptions (2023)  
 2103 [8] arXiv preprint: Agent-based approach for complex systems modelling (2023)  
 2104 [9] arXiv preprint: Information Filter upon Diversity-Improved Decoding (IFDID) for Natural  
 2105 Language Generation (2023)  
 2106 [6] arXiv preprint: Learning Optimal Pigovian Tax method (LOPT) for multi-agent reinforcement  
 2107 learning (2023)  
 2108 [2] arXiv preprint: International Taxation and its impact on Georgian Business Subjects (2023)  
 2109 [3] arXiv preprint: Assessing the reasoning depth of large language models (LLMs) (2024)  
 2110 [10] arXiv preprint: (Not provided, as it was not mentioned in the context)

2101  
 2102  
 2103  
 2104  
 2105

2097 Figure 15: Example of generated report by Storm for paper ‘TaxAgent: How Large Language Model  
 2098 Designs Fiscal Policy’.

2106  
 2107  
 2108  
 2109  
 2110  
 2111  
 2112  
 2113  
 2114  
 2115  
 2116  
 2117  
 2118  
 2119  
 2120  
 2121  
 2122

2123 **Box 8: Prompt to generate ArXiv Search Queries**

2124  
 2125 You are an expert technical writer generating targeted search queries to retrieve the most relevant arXiv  
 2126 papers for a technical report section.  
 2127 <Report topic>  
 2128 {{topic}}  
 2129 </Report topic>  
 2130 <Background>  
 2131 {{background}}  
 2132 </Background>  
 2133 <Task>  
 2134 Generate {number\_of\_queries} distinct arXiv search queries to comprehensively cover the section  
 2135 topic. Today's date is date.  
 2136 Guidelines for queries:  
 2137 1. Each query should use 1–10 keywords, focusing on a single, specific concept related to the topic.  
 2138 2. Ensure queries explore different or complementary aspects of the topic to maximize coverage.  
 2139 3. Use terminology and phrasing likely to match arXiv paper titles or abstracts.  
 2140 4. Avoid overly broad or generic queries; be as precise as possible.  
 2141 5. Queries should cover all the key aspects of the topic. Background information may be used to inform  
 2142 the queries.  
 2143 6. DO NOT create a complex query using AND/OR etc. Keep it simple  
 2144 The goal is to maximize the relevance and diversity of retrieved papers.  
 2145 </Task>

2146  
 2147  
 2148  
 2149  
 2150  
 2151  
 2152  
 2153  
 2154  
 2155  
 2156  
 2157  
 2158  
 2159

2160  
2161  
2162  
2163  
2164  
2165

2166 Box 9: Sem-Agg Instruction for final generation and summarization

2167

2168 You are an expert technical writer crafting one section of a technical report.

2169 <User Query>

2170 {topic}

2171 </User Query>

2172 <Section instructions>

2173 {section\_instructions}

2174 </Section instructions>

2175 <Existing section content (if populated)>

2176 {existing\_content}

2177 </Existing section content>

2178 <Source material>

2179 {context}

2180 </Source material>

2181 <Citation Guidelines>

2182 - Use [X] format where X is the {citation\_number}

2183 - Place citations immediately after the sentence or paragraph they are referencing (e.g., information from context [3]. Further details discussed in contexts [2][7].)

2184 - If urls are given in existing section content, rewrite them exactly if using information related to the url.

2185 - Make sure to provide citations whenever you are using information from the source material. This is a MUST.

2186 - Cite as many sources as possible.

2187 - Make sure to retain the citation numbers from the input context. - Provide in-line citations only. You do not need a reference section at the end.

2188 <Citation Guidelines>

2189 <Guidelines for writing>

2190 1. If the existing section content is populated, write a new section that enhances the existing section content with the new information. If not, write a new section from scratch.

2191 2. Provide groundings in the source material for all facts stated.

2192 3. When using information from a given source, make sure to cite the source.

2193 4. If a table or list would enhance understanding of a key point, and if so, include one.

2194 5. Make sure to follow the user query strictly.

2195 </Guidelines for writing>

2196 <Writing style>

2197 1. Content Requirements:

2198 - Ground all facts in the source material and provide citations.

2199 - Maintain an academic, technical focus throughout. No marketing language

2200 - Address potential counter-arguments where relevant.

2201 2. Structure and Formatting:

2202 - Use Markdown formatting.

2203 - Begin with ## for section title (Markdown format) and other headings as needed.

2204 - Use simple, clear language appropriate for academic writing.

2205 </Writing style>

2206 <Quality checks>

2207 - No preamble prior to creating the section content

2208 - Cite as many sources as possible.

2209 </Quality checks>

2210

2211

2212

2213