Improving Completeness in Deep Research Agents through Targeted Enrichment

Jesse Wonnink, Jakub Zavrel, Paul Groth
University of Amsterdam, Zeta Alpha
jessewonnink@gmail.com

Abstract

Deep research agents, AI systems that autonomously gather, synthesize, and report on complex topics, represent a significant advance in information synthesis, yet ensuring the completeness of their outputs remains an open challenge. A key bottleneck is query generation: current systems decompose research questions into subqueries via prompt engineering alone, offering no formal guarantees on diversity or coverage, which leads to redundant retrieval and gaps in the resulting reports. This paper presents HERO (High Enrichment Retrieval Orchestrator), a hierarchical deep research architecture that addresses this limitation through two complementary mechanisms. First, submodular optimization via a facility location objective provides mathematically grounded control over the relevance–diversity trade-off during query selection, replacing ad-hoc generation with provably diverse query sets. Second, a hierarchical enrichment stage independently analyzes each subquery pipeline's intermediate synthesis for information gaps and issues targeted follow-up queries, enabling adaptive depth without cross-pipeline interference. We evaluate HERO across academic (ScholarQABench) and general-domain (DeepResearchGym) benchmarks. HERO achieves state-of-the-art coverage, grounding, and presentation quality on DeepResearchGym, and the highest scores on multi-paper synthesis tasks in ScholarQABench.

Keywords: Deep Research Agents, Retrieval-Augmented Generation, Submodular Optimization, Report Completeness, Scientific NLP

1. Introduction

Deep Research (DR) agents are AI systems powered by Large Language Models (LLMs). They integrate dynamic reasoning, adaptive planning, and iterative tool use to acquire, aggregate, and analyze external information into comprehensive research reports (Huang et al., 2025) (Figure 1). Unlike task-specific Retrieval-Augmented Generation (RAG) applications (Singh et al., 2025), DR agents tackle open-ended research tasks where the scope and necessary depth are not predefined. They formulate search strategies, pursue follow-up questions based on initial findings, and synthesize information across multiple sources into structured outputs. As these systems become more capable and widely deployed, ensuring the *completeness* of their generated reports remains a central challenge (Huang et al., 2025). We operationalize completeness as the extent to which reports achieve three interdependent qualities: broad coverage of relevant aspects, factual grounding through verifiable citations, and coherent presentation.

A critical bottleneck lies in query generation. Current approaches decompose user questions into search queries through prompt engineering alone, providing no formal guarantees on diversity or coverage (Gao et al., 2023). This leads to redundant retrieval and incomplete reports, and existing systems lack principled mechanisms for identifying residual information gaps within their synthesized outputs.

This paper presents **HERO** (**H**igh **E**nrichment **R**etrieval **O**rchestrator), a hierarchical DR agent architecture that addresses these limitations through two complementary contributions. First, HERO employs **submodular optimization** for query selection by applying a facility location objective (Cornuéjols et al., 1977). This provides mathematically grounded control over the relevance–diversity trade-off in generated query sets, with approximation guarantees via greedy selection (Nemhauser et al., 1978). Second, HERO introduces a **hierarchical enrichment mechanism** in which each subquery pipeline independently analyzes its intermediate synthesis for information gaps and generates targeted follow-up queries to address them. This enables adaptive depth without cross-pipeline interference. These components are designed to be synergistic, jointly improving both breadth and depth of coverage.

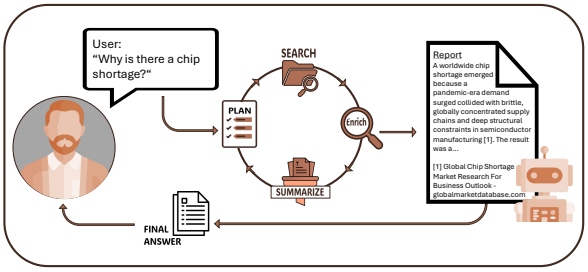

Figure 1: Deep research agent workflow: agents iteratively search, plan, synthesize, and enrich information before producing comprehensive reports.

Proceedings of the 3rd Int. Workshop on Natural Scientific Language Processing (NSLP 2026) @ LREC 2026, pages 193–205
12 May 2026.

We evaluate HERO on two complementary benchmarks: ScholarQABench (Asai et al., 2024), a multi-discipline scientific literature synthesis benchmark, and DeepResearchGym (Coelho et al., 2025), featuring complex web-based research queries evaluated against user click-derived ground truth. HERO achieves state-of-the-art performance on both benchmarks, attaining the highest Key Point Recall (67.63) and Citation F1 (91.57) on DeepResearchGym, and the strongest results on multi-paper ScholarQABench tasks (LLM-5: 4.95/5.0; ScholarQA-CS rubric score: 68.9 vs. 57.7 for the next-best baseline).

2. Related Work

2.1. Deep Research Agents

Deep Research agents extend agentic RAG frameworks by targeting open-ended research tasks that require comprehensive information gathering, multi-source synthesis, and structured report generation (Huang et al., 2025). Existing systems can be characterized along two principal dimensions (Huang et al., 2025). First, *workflow architecture* ranges from static pipelines with predetermined operation sequences to fully dynamic systems that adaptively decide when to continue or terminate exploration. OpenScholar (Asai et al., 2024) exemplifies the static approach through a fixed retrieve–rerank–generate pipeline with iterative refinement, while PaperQA2 (Skarlinski et al., 2024) operates dynamically, autonomously deciding when to search further or rewrite queries based on intermediate findings. Second, *agent composition* varies from single-agent systems where one LLM manages all decisions (Asai et al., 2024; Skarlinski et al., 2024) to multi-agent architectures where specialized components handle distinct subtasks. GPT-Researcher (Elovic, 2024) assigns parallel agents to research different subqueries simultaneously, and OpenDeepSearch (Alzubi et al., 2025) employs modular components for search and reasoning.

2.2. Query Diversification in Retrieval-Augmented Generation

Query decomposition is a well-established technique in RAG systems for handling complex information needs that cannot be satisfied by a single retrieval step (Singh et al., 2025). Early work by Perez et al. (2020) decomposed multi-hop questions into independently answerable subquestions, and a diverse array of methods has since emerged (Gao et al., 2023; Huang et al., 2023; Deng et al., 2022). In the context of DR agents, query decomposition serves a broader purpose: generating query sets that collectively cover the full scope of an open-ended research question. Yet current approaches rely on prompt engineering to encourage diversity, providing no formal guarantees about coverage or redundancy. This is particularly problematic for completeness, as queries must be sufficiently diverse to avoid redundant retrieval while remaining relevant to the original question. This is a trade-off that ad-hoc methods cannot explicitly control.

2.3. Submodular Optimization in Information Retrieval

Submodular optimization offers a principled framework for set selection under diversity constraints. The foundational result by Nemhauser et al. (1978) established that greedy maximization of monotone submodular functions achieves a $(1 - 1/e)$-approximation to the optimum, providing strong theoretical guarantees for practical algorithms. In information retrieval, submodular objectives have been successfully applied to *result* diversification: Agrawal et al. (2009) employed submodular functions to diversify web search results, and similar approaches have been adopted in recommender systems to balance accuracy with category diversity (Ashkan et al., 2015). The facility location objective (Cornuéjols et al., 1977) is particularly well-suited to relevance–diversity trade-offs, as it simultaneously maximizes coverage of a ground set while penalizing redundancy among selected items. Despite these successes in result diversification, submodular optimization has not previously been applied to query generation in DR agents. Concurrent work by Xiao (2025) explores related principles for query generation, though in a different architectural context. HERO applies submodular optimization at two levels: main query decomposition and enrichment query selection. To our knowledge, it is the first application of submodular optimization in deep research systems.

3. Method

We present HERO (High Enrichment Retrieval Orchestrator), a hierarchical multi-turn deep research architecture built around two core mechanisms: submodular optimization for query diversification and a targeted enrichment stage for adaptive depth. Figure 2 illustrates the complete system architecture.

3.1. System Overview

HERO operates in iterative research rounds, each consisting of query generation, parallel subquery execution, and enrichment. The process is governed by budget constraints: a maximum number of research turns $T_{\max}$, a per-turn subquery

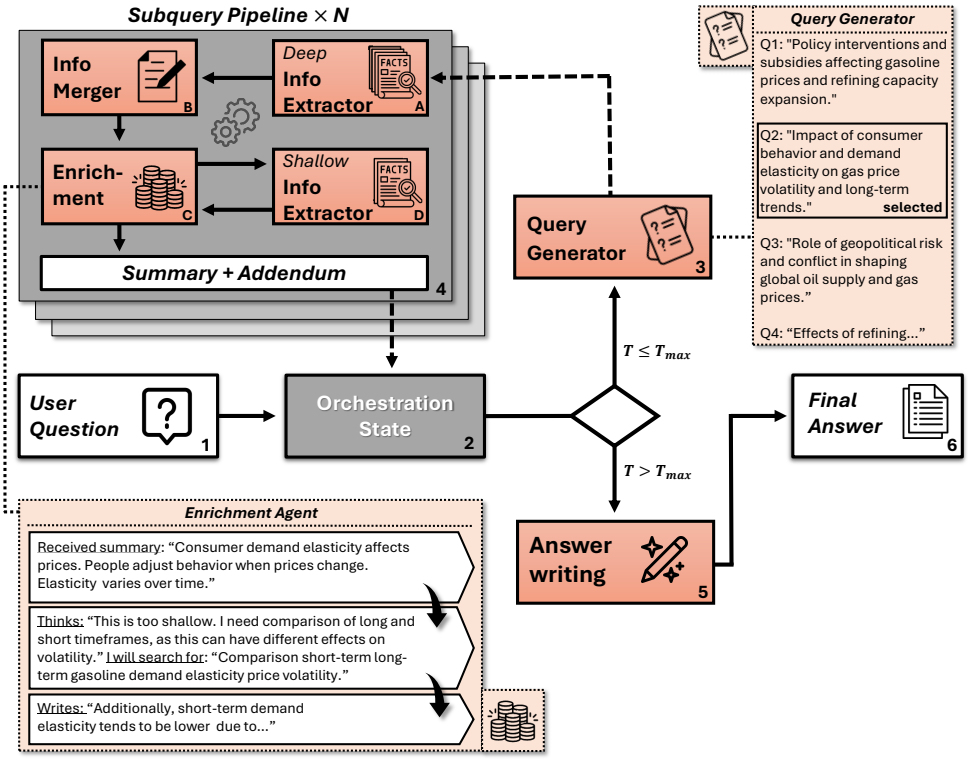

Figure 2: HERO system architecture. The orchestrator (steps 1–3) manages query generation, submodular selection, and answer synthesis. Each subquery pipeline (steps 4–6, stages A–D) independently performs information extraction, merging, enrichment analysis, and follow-up retrieval. Pipelines execute in parallel with isolated context to prevent cross-contamination.

budget k_t bounding the number of selected subqueries $|\mathcal{Q}_t| \leq k_t$, and a per-subquery enrichment budget k'_i bounding the number of follow-up queries $|\mathcal{Q}'_i| \leq k'_i$. These bounds ensure deterministic cost control while permitting adaptive exploration within each round. A central architectural principle is *hierarchical information isolation*: each subquery pipeline operates as an independent research branch, accessing only its own retrieved documents and intermediate summaries. After all rounds complete, the Answer Writer synthesizes all enriched summaries into the final report (Section 3.5).

3.2. Submodular Query Selection

A key limitation of existing DR agents is their reliance on prompt engineering for query decomposition, which provides no formal control over the diversity or relevance of generated queries. HERO addresses this through submodular optimization, applied at both the main query decomposition and enrichment query selection stages.

For each research turn t, the Query Generator produces a candidate pool V_t of $n = m \cdot k_t$ queries, where $m \geq 1$ is a candidate multiplier that ensures sufficient diversity for selection. From this pool, an optimal subset $\mathcal{Q}_t^* \subseteq V_t$ is selected by maximizing a facility location objective (Cornuéjols et al., 1977):

$$\mathcal{Q}_t^* = \arg \max_{\mathcal{Q}_t \subseteq V_t, |\mathcal{Q}_t| = k_t} f_\alpha(\mathcal{Q}_t) \tag{1}$$

where the objective function is defined as:

$$f_\alpha(\mathcal{Q}_t) = \sum_{j \in V_t} \max \Big(\alpha \cdot \text{sim}(\mathbf{e}_0, \mathbf{e}_j), \max_{q_i \in \mathcal{Q}_t} \text{sim}(\mathbf{e}_j, \mathbf{e}_i) \Big) \tag{2}$$

Here, $\mathbf{e}_j = \phi(q_j)$ denotes the embedding of candidate query q_j, $\mathbf{e}_0 = \phi(q_{\text{user}})$ is the embedding of the original user query, and sim denotes cosine similarity. The hyperparameter $\alpha \in [0, 1]$ governs the relevance–diversity trade-off: higher values of α anchor the selected queries closer to the user question, while lower values encourage broader exploration of the candidate space.

The function f_α is monotone submodular, which admits a greedy selection algorithm with a $(1-1/e)$-approximation guarantee (Nemhauser et al., 1978). This provides a principled alternative to ad-hoc query generation: rather than hoping the LLM produces diverse queries, HERO generates an intentionally large candidate set and then provably optimizes for coverage and diversity within it. Accordingly, the same optimization procedure is applied to select enrichment queries (Section 3.4).

In the first research round, the Query Generator receives only the user question. In subsequent rounds, it additionally receives previously executed subqueries and their enriched summaries, enabling the generation of candidates that target aspects not yet covered.

3.3. Subquery Pipeline

Each selected subquery $q_i \in \mathcal{Q}_t$ executes through an independent pipeline comprising information extraction and merging.

Information Extraction. The Information Extractor performs retrieval and relevance filtering for each subquery. It operates in two modes: *deep extraction* retrieves broadly relevant documents for the initial subquery, while *shallow extraction* performs targeted retrieval for enrichment queries with more specific search terms. Both modes employ the same extraction agent, which selects only relevant excerpts and factoids with accompanying source attribution. An early stopping criterion terminates retrieval when fewer than 10% of documents in a batch yield new citations, relative to the total sources already found. One additional low-yield batch is permitted before termination.

Information Merger. The Information Merger synthesizes all extracted information into a coherent intermediate research summary for the subquery. It connects disparate facts and excerpts into an information-dense paragraph, normalizing citations and removing redundant sources. Redundancy removal operates at the claim level: when two or more citations support overlapping or similar claims, the merger identifies the overlap and the LLM selects the strongest source to retain, consolidating the remaining citations. No substantive claims are discarded; only redundant source attributions are merged. This intermediate summary serves as the input for the Enrichment Agent and is ultimately consumed by the Answer Writer.

3.4. Hierarchical Enrichment

The second core contribution of HERO is a targeted enrichment mechanism that identifies and addresses information gaps in intermediate summaries through follow-up investigation. After the Information Merger produces an initial summary for subquery q_i, the Enrichment Agent analyzes it to determine whether meaningful gaps remain.

The Enrichment Agent receives four inputs: (1) the original user question, (2) the specific subquery focus, (3) the initial research summary from the Information Merger, and (4) the set of other subquery *topics* currently under investigation. This input structure is deliberately asymmetric: the agent has full access to its own pipeline's detailed findings but knows only the topics, not the content, of parallel pipelines.

Based on this analysis, the agent may generate up to k_i' enrichment queries targeting identified gaps. In practice, these gaps fall into three categories: *unexplored aspects* (e.g., a subquery on climate policy that covers economic impacts but lacks discussion of health effects), *insufficient evidence* (claims supported by only a single source where corroboration would strengthen the synthesis), and *unresolved contradictions* (conflicting findings across retrieved documents that require additional sources to reconcile). When multiple enrichment queries are proposed, submodular optimization (Equation 1) selects a diverse subset to prevent redundant follow-up searches. These enrichment queries are executed via shallow extraction. The newly retrieved information is then synthesized into an enrichment paragraph appended to the original summary, producing the *enriched summary*.

3.5. Answer Synthesis

The Answer Writer is the sole component with access to all enriched summaries across all research turns. It synthesizes these into a comprehensive final report by identifying cross-pipeline connections, resolving contradictions between independently gathered findings, and organizing information into a coherent narrative that directly addresses the user question. The Answer Writer produces the report along with a consolidated citation set drawn from all subquery pipelines.

4. Experimental Setup

We evaluate HERO on two complementary benchmarks: DeepResearchGym for general-domain research and ScholarQABench for scientific literature synthesis. For each benchmark, we organize metrics along three dimensions of completeness. *Coverage* assesses whether all relevant aspects of the research question are addressed; *grounding* measures factual accuracy and citation quality; *presentation quality* evaluates structural coherence and analytical depth. Evaluating all three jointly is necessary to assess report quality holistically, as high coverage without grounding produces unverifiable content, while strong grounding without adequate coverage leaves critical questions unanswered. Table 1 summarizes the full benchmark suite. For both benchmarks, baseline results are taken from the values reported by Coelho et al. (2025) and Asai et al. (2024), respectively.

Dataset	Format	Domain	N	Metrics
peS2o Corpus				
SciFact	Binary	Biomed	50	Corr, Cite
PubMedQA	Binary	Biomed	50	Corr, Cite
QASA	Short	CS	50	Corr, Cite
ScholarQA-CS	Long	CS	50	Corr, Cite
ScholarQA-BIO	Long	Biomed	50	Cite
ScholarQA-NEURO	Long	Neuro	50	Cite
ScholarQA-MULTI	Long	Mixed	50	Cite, LLM-5
ClueWeb22-B Corpus				
Researchy Qs	Long	General	100	KPR, Cite, LLM-10

Table 1: Benchmark suite composition. Corr: correctness (accuracy or ROUGE-L); Cite: citation F1; KPR: Key Point Recall; LLM-5/LLM-10: rubric-based quality scores on 5- and 10-point scales.

4.1. DeepResearchGym

We evaluate on the first 100 queries from the Researchy Questions dataset (Rosset et al., 2024), comprising complex, non-factoid information needs extracted from commercial search logs where users averaged 15.85 clicks across 6.31 unique documents per query. The retrieval corpus is ClueWeb22-B (Overwijk et al., 2022), containing 87 million English web documents indexed via MiniCPM-Embedding-Light (Hu et al., 2024) dense retrieval with DiskANN (Jayaram Subramanya et al., 2019) approximate nearest neighbor search, accessed through the retrieval API hosted by the DeepResearchGym authors.

We adopt the complete DeepResearchGym evaluation protocol, which employs `gpt-4.1-mini` (OpenAI, 2025) as judge (validated with $\kappa = 0.87$ human–LLM inter-annotator agreement, indicating strong alignment between automated and manual evaluation). *Coverage* is measured through Key Point Recall (KPR), the proportion of ground-truth key points (extracted from pages clicked by users in real search sessions) substantiated by the report, and Key Point Contradiction (KPC), the proportion contradicted. *Grounding* is assessed via Citation Recall (proportion of factual claims with $\geq$1 citation), Citation Precision (average support quality scored as 0, 0.5, or 1 per citation), and their harmonic mean F1. *Presentation quality* is evaluated through Clarity and Insightfulness, both on 0–10 scales. Implementation details regarding citation format adaptation and corpus retrieval adjustments are provided in Appendix D.

4.2. ScholarQABench

We evaluate on ScholarQABench (Asai et al., 2024), encompassing seven datasets across single-paper tasks (SciFact, PubMedQA, QASA) and multi-paper synthesis tasks (ScholarQA-CS, ScholarQA-BIO, ScholarQA-NEURO, ScholarQA-MULTI), using 50 queries per dataset. The retrieval corpus is peS2o v3 (Soldaini et al., 2024), comprising 45 million open-access scientific papers. We segment and encode this corpus into approximately 253 million chunks (mean length: 218.15 tokens, $\sigma = 65.10$) using a Contriever bi-encoder continually pre-trained on peS2o (Izacard et al., 2022).

Coverage metrics vary by task: accuracy for binary classification (SciFact, PubMedQA), ROUGE-L with BERTScore (Beltagy et al., 2019) for short-form generation (QASA), and expert-annotated rubric scores for ScholarQA-CS. For ScholarQA-MULTI, Prometheus V2 (Kim et al., 2024) evaluates coverage, organization, and relevance on a five-point Likert scale (LLM-5). *Grounding* is measured through Citation F1, where recall reflects the proportion of sentences ($\geq$50 characters) with NLI-verified support via Flan-T5-XL (Chung et al., 2022), and precision measures the proportion of necessary citations via leave-one-out ablation. Following Asai et al. (2024), we truncate to a maximum of three citations per sentence. We note that citation metrics are operationalized differently across benchmarks, precluding direct cross-benchmark comparison (see Appendix A).

4.3. HERO Configuration

Configuration parameters were adjusted based on task complexity and expected answer length. For long-form tasks (DeepResearchGym and ScholarQA-MULTI), HERO uses $T_{\max} = 3$ research turns with $k = 3$ subqueries per turn and $k' = 2$ enrichment queries per subquery. For ScholarQA-CS, we reduce to $T_{\max} = 2$ turns with $k' = 1$; remaining datasets use $T_{\max} = 2$, $k = 2$, $k' = 1$. Deep search depth is set to 24 documents per batch for web retrieval and 80 for academic retrieval, reflecting the smaller, more fragmented nature of peS2o chunks compared to full web pages. Shallow enrichment search uses one-third of these depths. Full per-dataset configurations are provided in Appendix C.

For submodular optimization, we set $\alpha = 0.6$ for main query generation and $\alpha = 0.65$ for enrichment queries, the latter prioritizing closer alignment with the subquery focus to prevent topic drift. Both use a candidate multiplier $m = 3$. These values were determined through preliminary experiments on the DeepResearchGym benchmark. Model assignments per agent are detailed in Appendix C.

Resource consumption. Table 8 (Appendix C) reports per-pipeline token costs. Each subquery pipeline consumes approximately 17k tokens on ClueWeb and 44k on peS2o, with the difference driven by deeper search over peS2o's more fragmented chunks. For a DeepResearchGym run with $k = 3$ subqueries per turn across $T_{\max} = 3$ turns, this amounts to approximately 9 subquery pipelines

at 17k each. The enrichment agent itself accounts for 1.3k tokens per pipeline on ClueWeb (1.8k on peS2o), while contributing a 3.6 percentage-point KPR improvement over the NoEnrichment ablation (Table 4). HERO is most beneficial for complex, multi-source synthesis tasks where comprehensive coverage justifies additional compute; for simple factoid or single-document tasks, lighter architectures may be more cost-effective.

4.4. Ablation Design

To isolate individual component contributions, we evaluate three configurations on a random sample of 20 DeepResearchGym queries: **HERO-Full** (both submodular optimization and enrichment), **NoEnrichment** (submodular optimization retained, $k' = 0$), and **NoSubmodular** (enrichment retained, queries selected via uniform random sampling from the candidate pool). All configurations use a single research turn ($T = 1$, $k = 4$, $m = 4$) to capture direct component effects before multi-turn iteration attenuates differences.

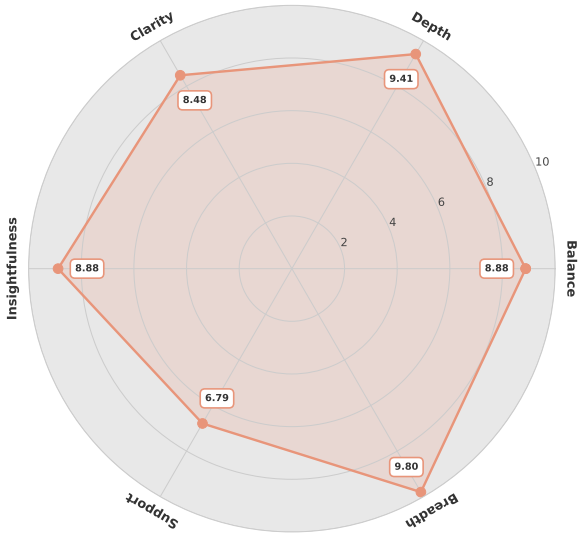

Figure 3: HERO's report quality across six dimensions (0–10) on DeepResearchGym. Breadth, Depth, Balance, and Support are not shown in Table 2; baseline scores for these dimensions were not reported in DeepResearchGym. Evaluated using GPT-4.1-mini.

5. Results

5.1. DeepResearchGym

Table 2 and Figure 3 present HERO's performance against five baselines on the general-domain benchmark.

Coverage. HERO attains the highest Key Point Recall at 67.63, surpassing the next-best system (GPT-Researcher, 64.67) by 2.96 percentage points. Figure 3 provides additional granularity across coverage-related dimensions.

Grounding. HERO achieves the highest Citation F1 (91.57), driven by near-perfect recall (99.35%) paired with competitive precision (84.92, slightly below GPT-Researcher's 85.36). HERO's Key Point Contradiction rate (2.34) is the highest among all systems; we analyze the sources of these contradictions in Section 7. The Support dimension on the radar chart (6.79) reflects HERO citing sources that the judge deemed insufficiently authoritative, which is an expected consequence of retrieving broadly without source quality filtering.

	Cov.	Grounding				Qual.	
System	KPR	KPC	Rec.	Prec.	F1	Clar.	Ins.
HERO	**67.63**	2.34	**99.35**	84.92	**91.57**	**8.48**	**8.88**
GPT-Researcher (Elovic, 2024)	64.67	1.42	90.82	**85.36**	87.96	8.37	7.80
OpenDeepSearch (Alzubi et al., 2025)	42.81	0.84	94.82	81.32	87.64	6.15	4.95
HF-DeepSearch (HuggingFace, 2025)	35.22	1.35	0.10	0.10	0.20	5.83	5.24
Search-o1* (Li et al., 2025)	29.93	**0.38**	–	–	–	3.03	3.79
Search-R1* (Jin et al., 2025)	4.95	0.80	–	–	–	0.91	1.12

Table 2: Performance on DeepResearchGym (Coelho et al., 2025) (first 100 queries). All metrics evaluated using GPT-4.1-mini as judge. Clarity and Insightfulness are on 0–10 scales. *Systems not tailored for long-report generation.

	Single-paper Datasets						Multi-paper Datasets					
	Pub		Sci		QASA		CS		Multi		Bio	Neu
Model	Corr	Cite	Corr	Cite	Corr	Cite	Corr	Cite	LLM-5	Cite	Cite	Cite
HERO	**100.0**	71.9	**100.0**	55.2	14.2 (86.4)	55.4	**68.9**	47.3	**4.95**	**60.5**	**66.9**	**72.6**
Llama3-8B	61.5	0.0	66.8	0.0	14.3	0.0	41.9	0.0	3.79	0.0	0.0	0.0
+OSDS	75.2	63.9	75.5	36.2	18.6	47.2	46.7	26.1	4.22	25.3	38.0	36.8
OS-8B	76.4	68.9	76.0	43.6	23.0	56.3	51.1	47.9	4.12	42.8	50.8	56.8
Llama3-70B	69.5	0.0	76.9	0.0	13.7	0.0	44.9	0.0	3.82	0.0	0.0	0.0
+OSDS	77.4	71.1	78.2	42.5	22.7	63.6	48.5	24.5	4.24	41.4	53.8	58.1
OS-70B	79.6	74.0	82.1	47.5	**23.4**	**64.2**	52.5	45.9	4.03	54.7	55.9	63.1
GPT4o	65.8	0.0	77.8	0.0	21.2	0.0	45.0	0.1	4.01	0.7	0.2	0.1
+OSDS	75.1	73.7	79.3	47.9	18.3	53.6	52.4	31.1	4.03	31.5	36.3	21.9
OS-GPT4o	74.8	**77.1**	81.3	**56.5**	18.7	60.4	57.7	39.5	4.51	37.5	51.5	43.5
PaperQA2 (Skarlinski et al., 2024)	–	–	–	–	–	–	45.6	**48.0**	3.82	47.2	56.7	56.0
Perplexity	–	–	–	–	–	–	40.0	–	4.15	–	–	–

Table 3: Performance on ScholarQABench (Asai et al., 2024) (50 samples per dataset). All baselines are from the OpenScholar evaluation. Corr: correctness (accuracy for PubMedQA/SciFact, ROUGE-L for QASA with BERTScore in brackets, rubric score for CS). Cite: citation F1. LLM-5: average of coverage, organization, and relevance as evaluated by Prometheus V2.

5.2. ScholarQABench

Presentation quality. HERO leads on both Clarity (8.48 vs. 8.37) and Insightfulness (8.88 vs. 7.80), the latter representing a margin of +1.08 points over the next-best baseline.

Table 3 presents results across seven scientific QA datasets.

Coverage. HERO achieves perfect accuracy (100%) on both PubMedQA and SciFact binary classification tasks, and the highest rubric score on ScholarQA-CS (68.9), outperforming the strongest baseline (OS-GPT4o, 57.7) by 11.2 points. QASA shows lower ROUGE-L (14.2) despite high semantic similarity (BERTScore 86.4), suggesting terminology mismatch rather than a substantive coverage gap. On ScholarQA-MULTI, HERO achieves near-ceiling performance with an LLM-5 score of 4.95 out of 5.0 (coverage: 4.88, organization: 5.0, relevance: 4.96), exceeding the average of all baselines (4.07) by 0.88 points.

Grounding. Citation F1 scores show a clear split by task type: HERO achieves the highest scores on multi-paper datasets (ScholarQA-MULTI: 60.5, ScholarQA-BIO: 66.9, ScholarQA-NEURO: 72.6, all best-in-class) while scoring mid-range on single-paper tasks. This pattern is consistent with HERO's architectural design for multi-source synthesis rather than single-document comprehension. After observing high single-paper correctness, we checked for potential data contamination. We identified 22 leaked chunks in SciFact, of which only 5 appeared in outputs; excluding them reduced F1 by 0.54 percentage points.

Presentation quality. The perfect organization score (5.0) on ScholarQA-MULTI directly reflects presentation quality, with HERO achieving the highest LLM-5 composite across all baselines.

5.3. Ablation Study

Table 4 presents the results of ablating HERO's two core components on a subset of 20 DeepResearchGym queries using a single research turn ($T = 1$, $k = 4$).

Both components contribute to coverage: disabling enrichment reduces KPR by 3.6 percentage points, while replacing submodular optimization with random selection reduces KPR by 5.3 percentage points. Neither ablated configuration matches the full system, confirming that the two mechanisms provide complementary rather than redundant improvements. The elevated standard deviations in both ablated conditions (21.0 and 21.1 vs. 16.7) suggest that the full system produces more consistently comprehensive reports across diverse query types.

Configuration	**KPR**
HERO-Full	**66.0** $\pm$ **16.7**
NoEnrichment	62.4 $\pm$ 21.0
NoSubmodular	60.7 $\pm$ 21.1

Table 4: Ablation results on DeepResearchGym (N=20, mean $\pm$ SD). NoEnrichment disables enrichment ($k' = 0$); NoSubmodular replaces submodular selection with uniform random sampling.

6. Conclusion

This paper addressed the challenge of ensuring completeness in deep research agents, where existing systems rely on ad-hoc query decomposition without formal guarantees about coverage or diversity. We presented HERO, a hierarchical architecture combining submodular optimization for principled query diversification with a targeted enrichment stage that identifies and addresses information gaps through follow-up investigation within each subquery pipeline. Evaluation across both academic (ScholarQABench) and general-domain (DeepResearchGym) benchmarks demonstrated state-of-the-art performance on coverage, grounding, and presentation quality. Ablation studies confirm that both components are necessary: neither alone matches the full system's coverage or consistency across diverse query types. These results demonstrate that principled optimization methods can be effectively integrated into agentic research workflows, offering measurable gains over prompt-engineering-based approaches to query generation. As deep research agents are increasingly deployed for scientific literature review and policy analysis, the architectural principles established here, namely formal diversification guarantees paired with adaptive hierarchical gap-filling, provide a concrete foundation for improving the reliability and comprehensiveness of AI-generated research reports. HERO is particularly valuable when exhaustive coverage matters: our results show it surfaces relevant information that other systems consistently miss, achieving the highest Key Point Recall across both benchmarks while maintaining strong grounding and presentation quality.

7. Limitations

Several limitations qualify the interpretation of our results.

Contradiction patterns and sycophantic bias. HERO's Key Point Contradiction rate (KPC = 2.34) is the worst among all baselines on DeepResearchGym. To investigate, we categorized all ground-truth key points by their stance relative to the query framing: *pro* (supporting the query's angle), *neutral*, or *anti* (opposing the framing). As shown in Figure 4, HERO is more than twice as likely to contradict key points opposing the query framing (3.42%) compared to supportive or neutral key points (1.53%), revealing systematic sycophantic bias. This manifests concretely as stance-flipping. When asked neutrally whether the death penalty should be legal, HERO supported all 7 anti-death-penalty key points while contradicting 4 of 8 pro-death-penalty key points. When the query was reframed as "why should the death penalty be allowed," HERO reversed position, supporting 10 of 12 pro key points while contradicting 2 of 4 anti key points. This pattern persists despite the Enrichment Agent's explicit instructions to identify contradictory evidence, demonstrating that architectural improvements alone cannot overcome behavioral tendencies inherent in foundation models (Sharma et al., 2025).

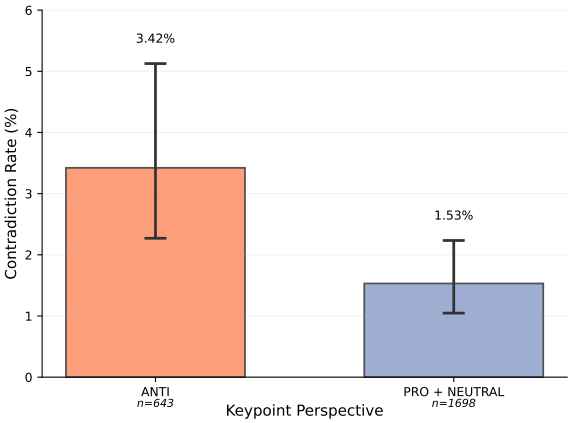

Figure 4: Contradiction rates by key point perspective on DeepResearchGym. Key points opposing the query framing (ANTI) are contradicted at more than twice the rate of supportive and neutral key points (PRO + NEUTRAL).

Not all contradictions stem from sycophancy. On the query "what role did Indians play in the wars for empire," HERO interpreted "Indians" as referring to people from the Indian subcontinent rather than Native Americans, producing a factually coherent but entirely misaligned report (KPR = 20%). Such semantic ambiguity failures represent a distinct failure mode from stance-dependent bias.

Foundation model advantage. HERO employs recent OpenAI models (`gpt-4.1-mini`, `gpt-5`) that are more capable than those available to baseline systems at the time of their evaluation. While the ablation study (Section 5.3) isolates architectural contributions within the same model family, a fully controlled comparison would require rerunning all baselines with equivalent foundation models. The extent to which absolute performance gains are attributable to architectural choices versus model improvements remains unclear.

Evaluation methodology. Both benchmarks rely on LLM-based evaluation, which introduces known biases. Research has documented a length bias in LLM judges, where longer responses receive systematically higher scores (Zheng et al., 2023). Since HERO is designed to produce comprehensive reports, its outputs are likely longer than base-

line outputs, potentially inflating rubric-based quality scores. Key Point Recall, which measures coverage against externally defined ground-truth points, is less susceptible to this confound. Additionally, citation metrics are operationalized differently across the two benchmarks: ScholarQABench's NLI-verified recall provides a stricter measure of citation quality than DeepResearchGym's presence-based check. This difference should be considered when interpreting the divergent grounding scores.

Citation task mismatch. HERO's citation F1 scores split markedly between single-paper and multi-paper ScholarQABench tasks (Table 3). As information passes through multiple synthesis stages, the system tends to select general descriptive chunks as citations rather than the specific passages that directly answer the query. Multi-paper datasets, the task type HERO is designed for, show stronger citation performance, while single-paper tasks expose this mismatch. A dedicated citation verification module (Qian et al., 2025) could mitigate this, particularly for synthesized claims aggregating evidence across sources.

Experimental design constraints. The ablation study evaluates 20 queries under a single-turn protocol ($T = 1$), which may favor breadth-oriented mechanisms (submodular optimization) over depth-oriented ones (enrichment), whose contributions likely compound across multiple rounds. The **NoEnrichment** configuration also performs strictly less computation, meaning the comparison measures effectiveness rather than efficiency. The submodular hyperparameters ($\alpha = 0.6$, $\alpha = 0.65$) were tuned on DeepResearchGym only; sensitivity analysis across domains was not conducted.

Domain scope. ScholarQABench covers computer science, biomedical, and neuroscience domains exclusively. DeepResearchGym draws from general web queries that can span a wider range of topics, including humanities and social sciences, though this diversity is not explicitly controlled or measured in the benchmark. While HERO's architecture is domain-agnostic (submodular optimization and hierarchical enrichment make no domain-specific assumptions), systematic evaluation on dedicated non-STEM benchmarks remains future work.

Model accessibility. HERO relies on commercial models (`gpt-4.1-mini`, `gpt-4o-mini`, `gpt-5`), which limits reproducibility. Evaluating HERO with open-weight models is a priority for future work.

8. Acknowledgements

This work was conducted as part of a Master's thesis at the University of Amsterdam, during an internship at Zeta Alpha, who provided computational resources and API costs for all experiments. We thank Jakub Zavrel for external supervision and Paul Groth for university supervision. Retrieval infrastructure for the peS2o corpus was hosted on the Snellius HPC cluster, accessed through SURF compute resources provided by the University of Amsterdam. We are grateful to the DeepResearchGym authors for providing updated API access to their retrieval infrastructure upon request.

9. Bibliographical References

Rakesh Agrawal, Sreenivas Gollapudi, Alan Halverson, and Samuel Ieong. 2009. Diversifying search results. In *Proceedings of the Second ACM International Conference on Web Search and Data Mining*, WSDM '09, pages 5–14, New York, NY, USA. ACM.

Salaheddin Alzubi, Creston Brooks, Purva Chiniya, Edoardo Contente, Chiara von Gerlach, Lucas Irwin, Yihan Jiang, Arda Kaz, Windsor Nguyen, Sewoong Oh, Himanshu Tyagi, and Pramod Viswanath. 2025. Open deep search: Democratizing search with open-source reasoning agents. *arXiv preprint arXiv:2503.20201*.

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

Azin Ashkan, Branislav Kveton, Shlomo Berkovsky, and Zheng Wen. 2015. Optimal greedy diversity for recommendation. In *IJCAI*, volume 15, pages 1742–1748.

Iz Beltagy, Kyle Lo, and Arman Cohan. 2019. SciBERT: A pretrained language model for scientific text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 3615–3620, Hong Kong, China. Association for Computational Linguistics.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Eric Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai,

Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Matthew D. Hoffman, Zoubin Ghahramani, Jakob Uszkoreit, Henryk Michalewski, Noam Shazeer, Patrick Li, Ed Chi, Quoc V. Le, and Denny Zhou. 2022. Scaling instruction-finetuned language models. *arXiv preprint arXiv:2210.11416*.

João Coelho, Jingjie Ning, Jingyuan He, Kangrui Mao, Abhijay Paladugu, Pranav Setlur, Jiahe Jin, Jamie Callan, João Magalhães, Bruno Martins, and Chenyan Xiong. 2025. Deepresearchgym: A free, transparent, and reproducible evaluation sandbox for deep research.

Gérard Cornuéjols, Marshall L. Fisher, and George L. Nemhauser. 1977. Location of bank accounts to optimize float: An analytic study of exact and approximate algorithms. *Management Science*, 23(8):789–810.

Zhenyun Deng, Yonghua Zhu, Yang Chen, Michael Witbrock, and Patricia Riddle. 2022. Interpretable amr-based question decomposition for multi-hop question answering. In *Proceedings of the International Joint Conference on Artificial Intelligence (IJCAI)*.

Assaf Elovic. 2024. Gpt-researcher. https://github.com/assafelovic/gpt-researcher. GitHub repository. Multi-stage agentic research pipeline for automated report generation. Accessed: 2025-10-12.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, and Haofen Wang. 2023. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*.

Shengding Hu, Yuge Tu, Xu Han, Chaoqun He, Ganqu Cui, Xiang Long, Zhi Zheng, Yewei Fang, Yuxiang Huang, Weilin Zhao, Xinrong Zhang, Zhi Thai, Kaihuo Zhang, Chongyi Wang, Yuan Yao, Weilin Zhao, Zhiyuan Liu, and Maosong Sun. 2024. Minicpm: Unveiling the potential of small language models with scalable training strategies. *arXiv preprint arXiv:2404.06395*.

Xiang Huang, Sitao Cheng, Yiheng Shu, Yuheng Bao, and Yuzhong Qu. 2023. Question decomposition tree for answering complex questions over knowledge bases. *arXiv preprint arXiv:2306.07597*.

Yuxuan Huang, Yihang Chen, Haozheng Zhang, Kang Li, Huichi Zhou, Meng Fang, Linyi Yang, Xiaoguang Li, Lifeng Shang, Songcen Xu, et al. 2025. Deep research agents: A systematic examination and roadmap. *arXiv preprint arXiv:2506.18096*.

HuggingFace. 2025. Open deep research.

Gautier Izacard, Edouard Grave, and Armand Joulin. 2022. Contriever: Unsupervised dense information retrieval with contrastive learning. *arXiv preprint arXiv:2112.09118*.

Suhas Jayaram Subramanya, Fnu Devvrit, Harsha Vardhan Simhadri, Ravishankar Krishnawamy, and Rohan Kadekodi. 2019. Diskann: Fast accurate billion-point nearest neighbor search on a single node. In *Advances in Neural Information Processing Systems*, volume 32. Curran Associates, Inc.

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

Seungone Kim, Juyoung Suk, Shayne Longpre, Bill Yuchen Lin, Jamin Shin, Sean Welleck, Graham Neubig, Moontae Lee, Kyungjae Lee, and Minjoon Seo. 2024. Prometheus 2: An open source language model specialized in evaluating other language models.

Xiaoxi Li, Guanting Dong, Jiajie Jin, Yuyao Zhang, Yujia Zhou, Yutao Zhu, Peitian Zhang, and Zhicheng Dou. 2025. Search-o1: Agentic search-enhanced large reasoning models. *arXiv preprint arXiv:2501.05366*.

George L. Nemhauser, Laurence A. Wolsey, and Marshall L. Fisher. 1978. An analysis of approximations for maximizing submodular set functions—I. *Mathematical Programming*, 14(1):265–294.

OpenAI. 2025. Introducing gpt-4.1 in the api. Accessed: 2025-10-13.

Arnold Overwijk, Chenyan Xiong, Xiao Liu, Cameron VandenBerg, and Jamie Callan. 2022. Clueweb22: 10 billion web documents with visual and semantic information.

Ethan Perez, Patrick Lewis, Wen-tau Yih, Kyunghyun Cho, and Douwe Kiela. 2020. Unsupervised question decomposition for question answering. *arXiv preprint arXiv:2002.09758*.

Haosheng Qian, Yixing Fan, Jiafeng Guo, Ruqing Zhang, Qi Chen, Dawei Yin, and Xueqi Cheng. 2025. Vericite: Towards reliable citations in retrieval-augmented generation via rigorous verification. In *Proceedings of the 2025 Annual International ACM SIGIR Conference on Research and Development in Information Retrieval in the Asia Pacific Region*, SIGIR-AP '25, pages 1–8, New York, NY, USA. ACM.

Corby Rosset, Ho-Lam Chung, Guanghui Qin, Ethan C. Chau, Zhuo Feng, Ahmed Awadallah, Jennifer Neville, and Nikhil Rao. 2024. Researchy questions: A dataset of multi-perspective, decompositional questions for llm web agents.

Mrinank Sharma, Meg Tong, Tomasz Korbak, David Duvenaud, Amanda Askell, Samuel R. Bowman, Newton Cheng, Esin Durmus, Zac Hatfield-Dodds, Scott R. Johnston, Shauna Kravec, Timothy Maxwell, Sam McCandlish, Kamal Ndousse, Oliver Rausch, Nicholas Schiefer, Da Yan, Miranda Zhang, and Ethan Perez. 2025. Towards understanding sycophancy in language models.

Aditi Singh, Abul Ehtesham, Saket Kumar, and Tala Talaei Khoei. 2025. Agentic retrieval-augmented generation: A survey on agentic rag. *arXiv preprint arXiv:2501.09136*.

Michael D Skarlinski, Sam Cox, Jon M Laurent, James D Braza, Michaela Hinks, Michael J Hammerling, Manvitha Ponnapati, Samuel G Rodriques, and Andrew D White. 2024. Language agents achieve superhuman synthesis of scientific knowledge. *arXiv preprint arXiv:2409.13740*.

Luca Soldaini, Rodney Kinney, Akshita Bhagia, Dustin Schwenk, David Atkinson, Russell Authur, Ben Bogin, Khyathi Chandu, Jennifer Dumas, Yanai Elazar, et al. 2024. Dolma: An open corpus of three trillion tokens for language model pretraining research. *arXiv preprint arXiv:2402.00159*.

Han Xiao. 2025. Submodular optimization for diverse query generation in deepresearch. Jina AI Tech Blog.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. 2023. Judging LLM-as-a-judge with MT-Bench and chatbot arena. In *Advances in Neural Information Processing Systems*, volume 36.

10. Language Resource References

João Coelho and Jingjie Ning and Jingyuan He and Kangrui Mao and Abhijay Paladugu and Pranav Setlur and Jiahe Jin and Jamie Callan and João Magalhães and Bruno Martins and Chenyan Xiong. 2025. *DeepResearchGym: A Free, Transparent, and Reproducible Evaluation Sandbox for Deep Research*. Carnegie Mellon University. PID https://arxiv.org/abs/2505.19253.

Akari Asai and Jacqueline He and Rulin Shao and Weijia Shi and Amanpreet Singh and Joseph Chee Chang and Kyle Lo and Luca Soldaini and Sergey Feldman and Mike D'arcy and others. 2024. *ScholarQABench*. Allen Institute for AI. PID https://github.com/AkariAsai/ScholarQABench.

Qiao Jin and Bhuwan Dhingra and Zhengping Liu and William Cohen and Xinghua Lu. 2019. *PubMedQA: A Dataset for Biomedical Research Question Answering*. University of Pittsburgh. PID https://pubmedqa.github.io/.

Yoonjoo Lee and Kyungjae Lee and Sunghyun Park and Dasol Hwang and Jaehyeon Kim and Hong-in Lee and Moontae Lee. 2023. *QASA: Advanced Question Answering on Scientific Articles*. KAIST. PID https://proceedings.mlr.press/v202/lee23n.html.

Arnold Overwijk and Chenyan Xiong and Xiao Liu and Cameron VandenBerg and Jamie Callan. 2022. *ClueWeb22: 10 Billion Web Documents with Visual and Semantic Information*. Carnegie Mellon University, Lemur Project. PID https://lemurproject.org/clueweb22/.

Corby Rosset and Ho-Lam Chung and Guanghui Qin and Ethan C. Chau and Zhuo Feng and Ahmed Awadallah and Jennifer Neville and Nikhil Rao. 2024. *Researchy Questions: A Dataset of Multi-Perspective, Decompositional Questions for LLM Web Agents*. Microsoft Research. PID https://arxiv.org/abs/2402.17896.

Luca Soldaini and Kyle Lo. 2023. *peS2o: Pretraining Efficiently on S2ORC Dataset*. Allen Institute for AI. PID https://huggingface.co/datasets/allenai/peS2o.

David Wadden and Shanchuan Lin and Kyle Lo and Lucy Lu Wang and Madeleine van Zuylen and Arman Cohan and Hannaneh Hajishirzi. 2020. *SciFact: A Dataset for Verifying Scientific Claims*. Allen Institute for AI. PID https://github.com/allenai/scifact.

A. Cross-Benchmark Citation Metrics

While both DeepResearchGym and ScholarQABench report citation recall and precision, these terms operationalize fundamentally different constructs (Table 5). DeepResearchGym's recall measures the *presence* of citations for LLM-extracted factual claims, while ScholarQABench's recall requires NLI-verified *support* for all sentences exceeding 50 characters. Precision differs similarly: DeepResearchGym evaluates support quality

with graduated scores (0, 0.5, 1), while ScholarQABench identifies *necessary* citations through leave-one-out ablation. These methodological differences preclude direct numerical comparison across benchmarks but provide complementary perspectives: DeepResearchGym incentivizes comprehensive attribution, while ScholarQABench rewards parsimonious citation.

Aspect	DeepResearchGym	ScholarQABench
Recall		
Scope	LLM-extracted factual claims	Sentences $\geq$50 chars
Validation	Presence check	NLI entailment
Precision		
Definition	Support quality	Citation necessity
Scoring	Graduated (0, 0.5, 1)	Binary (leave-one-out)
Philosophy	Comprehensive attribution	Parsimonious citation

Table 5: Citation metric operationalization across benchmarks.

B. ScholarQABench Citation Breakdown

Table 6 reports citation recall and precision independently for each ScholarQABench dataset. Multi-paper datasets show higher scores than single-paper tasks, consistent with HERO's architectural design for multi-source synthesis.

Dataset	Recall	Precision
Single-paper datasets		
PubMedQA	74.00	70.00
SciFact	56.00	54.33
QASA	55.17	55.67
Average	*61.72*	*60.00*
Multi-paper datasets		
ScholarQA-NEURO	74.92	70.47
ScholarQA-BIO	68.91	64.99
ScholarQA-MULTI	63.25	57.89
ScholarQA-CS	46.72	47.86
Average	*63.45*	*60.30*
Overall Average	**62.59**	**60.15**

Table 6: HERO's citation recall and precision per ScholarQABench dataset.

C. System Configuration Details

Table 7 reports the per-dataset configuration parameters. Average total token usage per subquery pipeline was approximately 17k tokens on ClueWeb and 44k on peS2o. Information extraction dominates cost on the academic corpus because peS2o's smaller, more fragmented chunks require deeper search. The system employs `gpt-4.1-mini` for query generation, information merging, and enrichment; `gpt-4o-mini` for information extraction; and `gpt-5` for answer writing.

Parameter	DRG	Multi	CS	Other†
Max Turns	3	3	2	2
Queries/Turn	3	3	3	2
Enrichments/Query	2	2	1	1
Deep Search Depth	24	80	80	80
Shallow Search Depth	8	40	40	40

Table 7: Agent configuration across datasets. DRG: DeepResearchGym; Multi: ScholarQA-MULTI; CS: ScholarQA-CS. †SciFact, PubMedQA, QASA, ScholarQA-BIO, ScholarQA-NEURO.

Agent	Model	ClueWeb	peS2o
Query Generator	gpt-4.1-mini	1.6	3.2
Info Extractor	gpt-4o-mini	2.2	7.3
Info Merger	gpt-4.1-mini	2.1	3.1
Enrichment	gpt-4.1-mini	1.3	1.8
Answer Writer	gpt-5	9.4	9.5
Total per subquery		**17**	**44**

Table 8: Average token usage per agent by corpus (thousands of tokens).

D. Evaluation Implementation Notes

Two adaptations were required for the DeepResearchGym evaluation. First, the claim-extraction prompt was adjusted to account for HERO's citation format, which places citations at sentence-end rather than inline with specific claims. Second, ClueWeb22-B's static corpus lacks URL-based retrieval, so the evaluation framework defaults to live web crawling, resulting in an approximately 40% document miss rate. Manual corpus search recovered 99.3% of cited documents in their snapshot state.

E. Enrichment Agent Prompt

The following is the system prompt for the Enrichment Agent (Section 3.4). The prompt operationalizes hierarchical information isolation: the agent receives parallel subquery *topics* via `completed_queries` but never the corresponding summaries.

```
{{ domain_prompt }}
{% if dataset_specific_instructions %}
# Dataset-Specific Guidance
{{ dataset_specific_instructions }}
{% endif %}

You are a research completeness analyst. Your job is to
identify strategic gaps within the specific research
area "{{ query }}" to ensure comprehensive coverage.

## MISSION
Analyze your research area and identify **0-
3 meaningful
gaps** that would significantly improve completeness.
Focus on YOUR topic, not the overall question.

## COMPLETENESS ASSESSMENT
**Today's Date:** {{ today_date }}

### Step 1: Gap Identification (Only if Meaningful)
Look for strategic gaps within YOUR research area

### Step 2: Targeted Enrichment (If Gaps Found)
For each valid gap, create ONE specific search query:
- **Specificity**: Target exact aspect missing
- **Searchability**: Use concrete terms likely in docs
- **Value**: High chance of substantial new information
- **Write relatively short and simpler queries that are
  easy to understand and execute, be more specific in
  the extraction instructions.**

## RESEARCH STATUS
**Overall Context:** {{ original_question }}
**Your Focus Area:** {{ query }}
{{ merged_information }}

{% if completed_queries %}
## PARALLEL RESEARCH
{% for completed_query in completed_queries %}
- "{{ completed_query }}"
{% endfor %}
Stay within your research area only.
{% endif %}

{% if round_num > 1 %}
## ROUND {{ round_num }} - REFINE, DON'T REPEAT
Previous round targeted gaps but some may persist.
If so:
- Generate MORE SPECIFIC queries (countries,
  mechanisms, timeframes, stakeholders)
**Refinement pattern**: "policy impact" ->
  "California EV policy impact 2024"
{% endif %}

## DECISION LOGIC
**STOP enrichment** if:
- Topic area is thoroughly covered from key angles
- Additional searches would likely be redundant
- No meaningful gaps remain within your scope

**CONTINUE enrichment** if:
- Clear strategic gaps exist within your research area
- Missing information would meaningfully improve
  topic completeness

## OUTPUT FORMAT (JSON)
**If meaningful gaps exist**: Generate EXACTLY
{{ max_candidate_queries }} enrichment tasks.
**If NO meaningful gaps exist**: Return empty array.

**Final Check**: Only propose enrichments that would add
meaningful, substantial content to your specific
research area.
```