HAWKBENCH: Investigating Resilience of RAG Methods on Stratified Information-Seeking Tasks

Hongjin Qian[1,2], **Zheng Liu**[2,3]*, **Chao Gao**[6], **Yankai Wang**[5]
Defu Lian[5], **Zhicheng Dou**[4]
[1] Peking University
[2] Beijing Academy of Artificial Intelligence
[3] Hong Kong Polytechnic University
[4] Renmin University of China
[5] University of Science and Technology of China
[6] Hong Kong University of Science and Technology
{chienqhj,zhengliu1026}@gmail.com

Abstract

In real-world information-seeking scenarios, users have dynamic and diverse needs, requiring RAG systems to demonstrate adaptable resilience. To comprehensively evaluate the resilience of current RAG methods, we introduce HawkBench, a human-labeled, multi-domain benchmark designed to rigorously assess RAG performance across categorized task types. By stratifying tasks based on information-seeking behaviors, HawkBench provides a systematic evaluation of how well RAG systems adapt to diverse user needs. Unlike existing benchmarks, which focus primarily on specific task types (mostly factoid queries) and rely on varying knowledge bases, HawkBench offers: (1) systematic task stratification to cover a broad range of query types, including both factoid and rationale queries, (2) integration of multi-domain corpora across all task types to mitigate corpus bias, and (3) rigorous annotation for high-quality evaluation. HawkBench includes 1,600 high-quality test samples, evenly distributed across domains and task types. Using this benchmark, we evaluate representative RAG methods, analyzing their performance in terms of answer quality and response latency. Our findings highlight the need for dynamic task strategies that integrate decision-making, query interpretation, and global knowledge understanding to improve RAG generalizability. We believe HawkBench serves as a pivotal benchmark for advancing the resilience of RAG methods and their ability to achieve general-purpose information seeking. We host our codes and data in *this repository*.

1 Introduction

Large Language Models (LLMs) excel in general reasoning and knowledge-based tasks but often struggle with timeliness and knowledge coverage gaps, particularly in specialized domains and user-specific data [OpenAI, 2023, DeepSeek-AI, 2024]. To address these limitations, incorporating external knowledge has become a common approach, with Retrieval-Augmented Generation (RAG) emerging as an effective solution to enhance factual accuracy and adaptability [Zhu et al., 2024].

During the information-seeking process using RAG, users may have a wide range of information needs, from simple factoid retrieval to more complex rationale-based queries [Qian et al., 2025, Zhao et al., 2024a]. This versatility requires RAG systems to possess diverse capabilities, including accurate referencing and advanced reasoning skills.

*Corresponding author.

39th Conference on Neural Information Processing Systems (NeurIPS 2025).

Recent advancements in RAG methods have enhanced vanilla RAG systems by targeting specific advanced capabilities. For instance, some methods focus on improving multi-hop reasoning to handle tasks with implicit information intents [Zhao et al., 2024b, Xu et al., 2024], while others address information aggregation tasks by constructing intermediate structures, such as graphs or memory modules, to better integrate relevant information [Qian et al., 2025, Edge et al., 2024].

While these advancements enable RAG systems to effectively leverage external knowledge for specific tasks, their ability to generalize across diverse scenarios remains uncertain. A recent survey categorizes external knowledge-based tasks into distinct levels, emphasizing that no single method can effectively address all query types [Zhao et al., 2024a]. This suggests that current RAG methods lack the resilience required for general-purpose information-seeking tasks, highlighting the need for a systematic evaluation of RAG methods across a broad range of information-seeking tasks, examining the resilience of these methods when faced with information-seeking tasks in any form.

Existing public benchmarks for RAG evaluation focus narrowly on isolated dimensions of information-seeking tasks. For instance, LegalBench-RAG evaluates information-seeking tasks in the legal domain [Guha et al., 2023], MutiHop-RAG tests multi-hop reasoning [Tang and Yang, 2024], and CRAG emphasizes comprehensive evaluation on factual QA tasks [Yang et al., 2024]. While these benchmarks excel in their targeted domains, they collectively fail to assess the resilience of RAG methods across stratified task types due to three critical limitations:

First, fragmented evaluation protocols. Current benchmarks are siloed by design, each prioritizing distinct query types. This specialization creates inconsistent evaluation criteria, hindering fair comparisons of RAG performance across diverse task categories. **Second, domain bias and knowledge leakage**. Many benchmarks rely on heterogeneous knowledge bases (e.g., Wikipedia and web snippets), leading to corpus-dependent performance gaps that obscure true method capabilities. Worse, LLMs are often pretrained on these same sources (e.g., Wikipedia), inflating benchmark scores through memorization rather than genuine retrieval-augmented reasoning. **Third, limited query diversity.** Most benchmarks disproportionately emphasize factoid questions (e.g., "When was Einstein born?"), neglecting rationale-based queries (e.g., "Explain how relativity revolutionized physics") that require synthesis and contextual analysis. This narrow focus misaligns with real-world user needs, where information-seeking behaviors span both factual lookup and complex reasoning.

HawkBench is characterized by the following key features:

Domain Thoroughness – We curate raw texts from a diverse range of sources—including professional textbooks, academic papers, financial reports, legal contracts, and novels—to ensure that the benchmark reflects real-world information needs. This broad selection captures both general and specialized knowledge, offering a robust foundation for evaluation.

Systematic Task Stratification – We systematically define four query types: (1) explicit factoid queries, (2) implicit factoid queries, (3) explicit rationale queries, and (4) implicit rationale queries. This stratification, inspired by Zhao et al. [2024a] with refined modifications, ensures comprehensive task coverage. Importantly, all query types share the same underlying knowledge distribution, allowing for direct and fair performance comparisons across different tasks.

Rigorous Annotation Quality – HawkBench employs a hybrid annotation process that leverages both advanced LLMs—specifically GPT-4 and DeepSeek-V3—and human oversight. Initially, LLMs generate query-answer pairs from the curated texts. Expert annotators then evaluate these pairs against predefined stratification levels, refine the answers by correcting inaccuracies, and enhance clarity. This process results in a high-quality dataset of 1,600 annotated test samples, evenly distributed across all task types.

We further validate HawkBench by applying representative RAG methods and performing a comprehensive analysis of their performance in terms of both answer quality and response latency. Our empirical results reveal that while current RAG methods excel in specific tasks, they generally lack overall resilience. Enhancing their adaptability will require dynamic task strategies that integrate decision-making, query interpretation, and a holistic understanding of global knowledge.

Our contributions are as follows: (1) We introduce HawkBench, a high-quality benchmark with stratified tasks designed to assess the resilience of RAG methods for general-purpose information-seeking. (2) We conduct a comprehensive empirical evaluation of recent RAG methods on HawkBench,

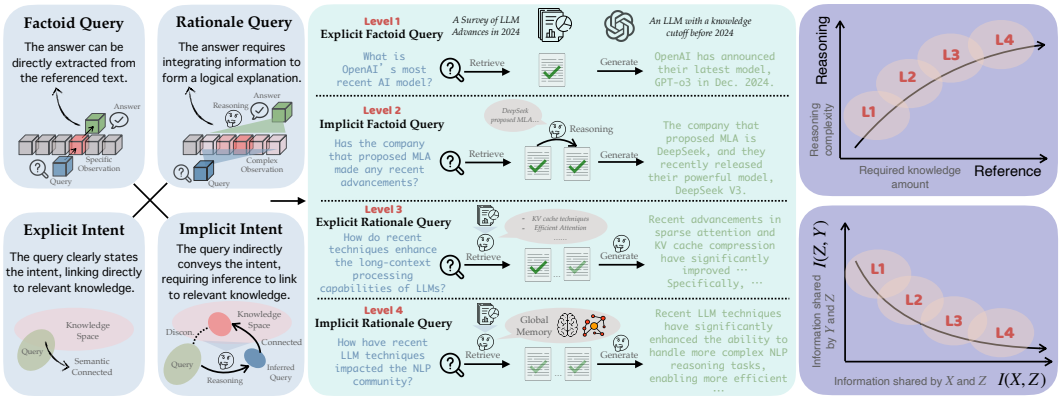

Figure 1: Query Stratification of HAWKBENCH. To account for referencing difficulty, we categorize tasks into queries with explicit intent and implicit intent. Regarding reasoning, tasks are categorized into factoid queries and rationale queries. By combining these two categorizations, we stratify information-seeking tasks into four levels.

enabling a side-by-side comparison of their capabilities. (3) We propose insights and strategies to improve the generalizability and adaptability of current RAG methods.

2 HawkBench

2.1 Preliminary

Recent advancements in large language models (LLMs) have popularized the Retrieval-Augmented Generation (RAG) approach, which leverages external knowledge to perform specific tasks. In RAG, a generation model $\theta(\cdot)$ and a retrieval model $\gamma(\cdot)$ collaborate to produce a final response $\mathcal{Y}$. Formally, the process is expressed as:

$$\mathcal{Y} = \theta(q, \mathcal{Z}), \quad \mathcal{Z} = \gamma(q, \mathcal{X}), \tag{1}$$

where q denotes the input query, $\mathcal{X}$ represents the external knowledge base, $\mathcal{Z}$ is the retrieved relevant information, and $\mathcal{Y}$ is the generated answer.

This RAG framework can be viewed as an information-refinement process following the *Markov chain*: $\mathcal{X} \to \mathcal{Z} \to \mathcal{Y}$. As information passes through each stage, it is progressively distilled, leading to the inequality $I(\mathcal{X}, \mathcal{Z}) \geq I(\mathcal{Y}, \mathcal{Z})$, where $I(\cdot)$ denotes mutual information. Ideally, the retrieval step should extract a $\mathcal{Z}$ that is both *sufficient*—containing all the information necessary to generate $\mathcal{Y}$—and *minimal*—excluding irrelevant details from $\mathcal{X}$. In fact, the condition $I(\mathcal{X}, \mathcal{Z}) = I(\mathcal{Y}, \mathcal{Z})$ would hold if and only if an optimal retrieval output $\mathcal{Z}^*$ exists that perfectly balances these two criteria. Achieving such an optimal $\mathcal{Z}^*$ is challenging due to estimation biases in both the retrieval and generation processes. To better understand these challenges, it is essential to consider two interrelated dimensions:

Referencing The retrieval process must determine not only which pieces of information in $\mathcal{X}$ are relevant to the query q but also how much information is required. The *referencing* is straightforward when q explicitly states its intent, as the semantic connections between q and the relevant content in $\mathcal{X}$ are easier to measure. However, for implicit queries—where the intent is not clearly stated—identifying the necessary evidence becomes more complex. Thus, the referencing dimension measures *how to access* the relevant knowledge, capturing both the volume of information needed and its accessibility within the knowledge base.

Reasoning Once the retrieval model produces $\mathcal{Z}$, the generation model must process and integrate this information to formulate the final answer $\mathcal{Y}$. For factoid queries, the retrieved information typically aligns closely with the required answer, meaning that the reasoning effort is relatively minimal. In contrast, when the query demands a rationale—requiring the synthesis and integration of multiple pieces of information—the generation process must engage in more complex in-context

reasoning. Therefore, the reasoning dimension measures *how to utilize* the relevant knowledge, reflecting the cognitive effort needed to bridge the gap between the retrieved data and the final, coherent response.

To systematically analyze the difficulty of information-seeking tasks within the RAG framework, we decompose queries along these two dimensions. As shown in Figure 1 (left), we categorize tasks based on: **Referencing:** Whether the query explicitly or implicitly conveys its intent, thereby affecting the ease with which relevant information can be identified. **Reasoning:** Whether the task involves straightforward fact extraction or requires integrating information to form a reasoned response. By combining these dimensions, we define four levels of information-seeking tasks, each posing unique challenges to the RAG pipeline, as outlined in the next section.

2.2 Query Stratification

In Figure 1 (middle), we illustrate our query stratification, presenting the four query types below.

Level 1: Explicit Factoid Query Level 1 queries exhibit an explicitly stated information-seeking intent and typically require minimal reasoning. The answer is directly available in the retrieved text. For instance, the query

> *"What is OpenAI's most recent AI model?"*

clearly specifies its intent, allowing the retrieval system to easily locate the pertinent information. The generator can then extract the final answer with little or no additional reasoning.

Level 2: Implicit Factoid Query Level 2 queries present an implicit information-seeking intent, which necessitates an extra step to resolve the reference before the answer can be extracted. Consider the query

> *"Has the company that proposed MLA made any recent advancements?"*

The query does not directly name the company. The system must first infer that "the company that proposed MLA" refers to, for example, DeepSeek. Once this implicit reference is established, the relevant knowledge can be retrieved, and the answer can be extracted with minimal reasoning. Thus, Level 2 queries require additional referencing effort compared to Level 1, while the reasoning for answer extraction remains straightforward.

Level 3: Explicit Rationale Query In Level 3 queries, the intent is explicitly stated, but there exists a semantic gap between the query and the relevant information. Although the query clearly indicates what is being asked, the final answer is not directly extractable from a single text fragment and requires synthesizing information from multiple sources. For example, the query

> *"How do recent techniques enhance the long-context processing capabilities of LLMs?"*

explicitly requests an explanation. However, the necessary rationale is dispersed across several texts. This scenario demands a more complex retrieval process, possibly aided by structured representations (e.g., graphs), and a generator capable of synthesizing the information into a coherent answer.

Level 4: Implicit Rationale Query Level 4 queries pose the highest challenge as they involve both an implicit intent and the need to generate a global explanation. For example, the query

> *"How have recent LLM techniques impacted the NLP community?"*

requires the system to first infer the underlying intent and then integrate diverse pieces of information across the entire knowledge base to form a comprehensive explanation. This task demands extensive referencing to identify loosely connected yet relevant content and significant reasoning to synthesize a unified, high-level response.

2.3 Comparison of the Four Query Levels

In Figure 1 (right), we compare the four query levels across two aspects: *Reference* and *Reasoning*. First, in terms of *Reference*, the amount of relevant knowledge required increases from Level 1

to Level 4 queries, reflected in the mutual information between the knowledge base and retrieved knowledge, $I(\mathcal{X}, \mathcal{Z})$. Level 1 queries require minimal knowledge, as answers are directly extractable from a few text chunks. In contrast, higher-level queries, such as Level 3 and Level 4, require synthesizing information from a broader range of texts. Second, in terms of *Reasoning*, complexity increases across levels due to the growing semantic gap between retrieved knowledge and the final answer. For Level 1 queries, reasoning is minimal, but for Level 3 and Level 4 queries, more reasoning is needed to connect multiple, loosely connected pieces of information. This is reflected in the decreasing mutual information $I(\mathcal{Z}, \mathcal{Y})$ as redundant information is filtered out during refinement.

These varying requirements for referencing and reasoning present significant challenges for current RAG systems, which struggle to adapt to the diversity of information-seeking tasks. There is no one-size-fits-all solution, as each task demands distinct capabilities. This underscores the necessity of benchmarking current RAG methods across a broad range of tasks to better assess their resilience.

2.4 Construction

Corpus Collection While most current LLMs are proficient in general world knowledge due to their training on large-scale corpora, they often lack coverage in specialized, domain-specific areas. To address this gap, HawkBench incorporates 229 domain-specific texts into its knowledge base. These texts are carefully selected from a larger collection of long texts gathered across diverse domains, which can also serve as a global corpus for retrieval. The selected 229 contexts span a wide range of domains, including professional textbooks (manually labeled into categories such as technology, humanities, art, and science), financial reports, legal contracts, novels, and academic papers. This diverse and comprehensive collection ensures that HawkBench can thoroughly evaluate the domain resilience of RAG methods by covering a broad range of user information needs.

Annotation Process The annotation process for constructing HawkBench follows a systematic approach, as illustrated in Figure 6. The process consists of three key steps:

(1) **Configuration:** The annotator selects the target query level and domain, with assistance from a strong LLM (GPT-4o and DeepSeek-v3).

(2) **Question-Answer Pair Generation:** The system prompts the LLM agent using built-in QA generation prompts to produce initial question-answer pairs. During this step, the system first samples from the knowledge base, selecting a random text span of varying lengths based on the task type. For Level-1 tasks, approximately 1K tokens are used as the context. For Level-2 tasks, we use a retrieval system retrieves the top-10 passages based on the generated L1 query, selecting five passages to prompt the agent to transform explicit factoid queries into implicit intent queries. For Level-3 and Level-4 tasks, up to 120K tokens are sampled as the knowledge context to guide the agent in generating information aggregation queries, with different prompts controlling the process. The codes for annotation system and all built-in prompts are in *this repository*.

(3) **Quality Control:** The annotator reviews the generated question to ensure it aligns with the target task type's definition. If the question is unsuitable, it is discarded. If the question is valid, the annotator evaluates the generated answer for clarity, conciseness, and semantic richness. The answer is then manually edited to ensure high quality.

L	Discard %	Edit %	Ave. Time	Total Time
1	6.7%	3.5%	26s	4.5h
2	28.1%	41.4%	71s	23.1h
3	25.2%	47.9%	183s	41.5h
4	29.1%	40.6%	201s	45.2h

Figure 2: Statistical Details of Construction.

We employed three PhD students proficient in English as annotators. As shown in Table 2, the difficulty of annotating different task types varies significantly. For Level-1 tasks, most generated QA pairs are valid with only minor edits needed, making this task relatively quick. In contrast, for Levels 2–4, the generated QA pairs are often invalid and discarded, and the quality of the answers generally requires more extensive manual editing. This process results in longer annotation times for higher-level tasks. The total annotation time includes both system latency (primarily due to QA pair generation) and manual annotation work. The three annotators dedicated approximately one week of full-time work to constructing HawkBench, each receiving a salary of around $1,000. Additionally, constructing HawkBench incurred around $597 in GPT-4o usage and $278 in DeepSeek-v3 usage.

Dataset Distribution Table 3 presents the statistical details of HawkBench. The dataset contains 1,600 test samples, derived from 229 context knowledge bases. The compressed file size of HawkBench is only 26MB, making it highly portable for distribution. We have thoroughly reviewed the licenses of all source texts to ensure that they permit redistribution. HawkBench is distributed under the Apache License 2.0.

3 Experiment

3.1 Baselines and Metrics

To investigate the resilience of RAG methods on HawkBench, we select the following representative baseline methods: **Vanilla RAG:** This method retrieves the top passages as context. **Enhanced RAG Methods:** *HyDE* [Gao et al., 2023] generates a hypothetical document to enhance query retrieval. *RQRAG* [Chan et al., 2024] rewrites the input query into sub-queries to refine retrieval. **Global RAG:** These methods index the knowledge base into an intermediate form to enhance global awareness. This includes memory-based methods such as *MemoRAG* [Qian et al., 2025] and graph-based methods like *GraphRAG* [Edge et al., 2024].

Additionally, we explore the application of long LLMs in HawkBench, including vanilla LLMs, the prompt compression method *Lingua-2* [Pan et al., 2024], and long-context acceleration methods such as *MInference* [Jiang et al., 2024]. All baselines in the main experiments use *Qwen2.5-7B-instruct* as the generator [Team, 2025], with *BGE-M3* as the retriever [Chen et al., 2023] and the top-k set to 5 for all RAG methods.

For Level 1 and Level 2 tasks, which focus on factoid queries, we use *Rouge-L* and lexical F1-score as evaluation metrics. These metrics emphasize surface-form lexical overlap and are well suited for evaluating fact-based answers.

For Level 3 and Level 4 tasks, which involve rationale queries, we introduce a new evaluation metric, denoted as $\mathcal{S}$-F1, to robustly assess sentence-level semantic equivalence between the ground-truth and predicted answers. Specifically, let A^* denote the ground-truth answer and A the predicted answer. We tokenize both A and A^* into sentences $\{s_i\}$ and $\{s_i^*\}$, respectively. Then, $\mathcal{S}$-F1 is defined as:

$$\mathcal{S}\text{-F1}(A, A^*) = \frac{1}{2n}\sum_{i=1}^{n} \mathbb{1}_{\{\text{LLM}(s_i, A^*)=\text{True}\}} + \frac{1}{2m}\sum_{i=1}^{m} \mathbb{1}_{\{\text{LLM}(s_i^*, A)=\text{True}\}}, \tag{2}$$

where $\mathbb{1}_{\text{condition}}$ is an indicator function that returns 1 if the condition holds and 0 otherwise, n is the number of sentences in A, and m is the number of sentences in A^*.

Intuitively, $\mathcal{S}$-F1 computes the average of: (1) **Precision:** the proportion of sentences in the predicted answer $s_i \in A$ that are judged by a strong LLM to be semantically supported by the ground-truth answer A^*. (2) **Recall:** the proportion of sentences in the ground-truth answer $s_i^* \in A^*$ that are judged to be semantically supported by the predicted answer A.

Here, "supported by" means that for a given sentence, the LLM judge determines whether its meaning or rationale is present, possibly rephrased but semantically equivalent, in the other answer. More concretely, we apply the following process: (1) Each predicted answer is split into sentences. For each $s_i \in A$, the LLM judge is prompted to return a binary decision (0/1) indicating whether the content of s_i is covered by A^*. (2) Each ground-truth answer is similarly split into sentences, and for each $s_i^* \in A^*$, we query whether its rationale is reflected in A.

Compared to lexical F1-score, $\mathcal{S}$-F1 moves beyond surface-form matching and directly evaluates sentence-level semantic alignment between A and A^*, making it a more robust metric for rationale-based tasks where lexical overlap alone cannot capture equivalence. For completeness, we also report *Rouge-L* scores alongside $\mathcal{S}$-F1 when evaluating Level 3 and Level 4 tasks.

3.2 Main Results

We conduct comprehensive experiments across all baselines, with the full results presented in Table 5. To provide a more detailed analysis, we examine the results from multiple perspectives, offering a deeper understanding of performance across different dimensions.

Table 1: Evaluation performance across four levels, averaged over all domains. The best scores are highlighted in bold, and the second-best scores are underlined.

Method	Type	LEVEL-1		LEVEL-2		LEVEL-3		LEVEL-4	
		Rouge-L	F1	Rouge-L	F1	Rouge-L	$\mathcal{S}$-F1	Rouge-L	$\mathcal{S}$-F1
LLM	Long LLM	13.0	12.9	12.9	11.5	<u>26.2</u>	24.0	16.9	33.2
Lingua-2	Compression	11.4	11.4	12.2	11.4	23.7	23.9	15.4	25.2
MInference	Accelerating	11.5	11.1	12.6	11.2	25.6	24.2	17.1	<u>33.3</u>
RAG	Standard RAG	50.9	57.5	34.0	38.6	17.9	27.3	15.3	18.3
HyDE	Enhanced RAG	**64.4**	<u>73.5</u>	<u>40.0</u>	<u>44.5</u>	19.4	28.0	15.6	18.4
RQRAG	Enhanced RAG	<u>64.2</u>	**73.6**	**41.1**	**46.8**	19.7	28.6	15.4	17.4
MemoRAG	Global RAG	44.8	50.2	33.7	37.3	**27.3**	**34.1**	<u>19.0</u>	**35.0**
GraphRAG	Global RAG	49.3	57.4	34.0	37.0	25.3	<u>32.5</u>	**20.6**	28.7

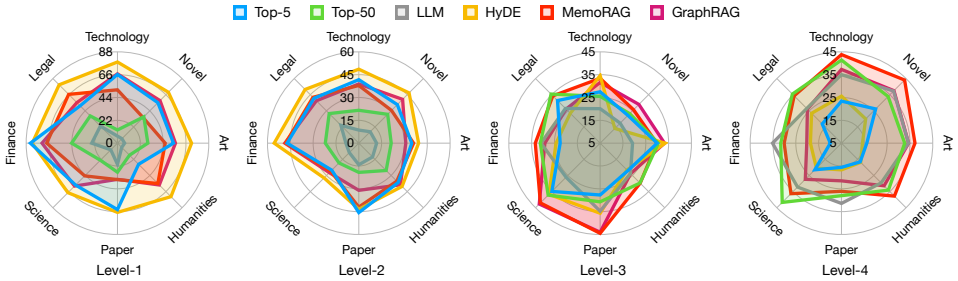

Figure 3: Evaluation performance across four levels and eight domains for selected methods.

Resilience across Levels Table 1 presents the performance of all baselines across the four task levels, averaged by domain. From these results, we draw several key insights: (1) Standard RAG and Enhanced RAG methods perform well on factoid queries (Level-1 and Level-2), suggesting that these queries often rely on specific text spans that can be easily located with minimal reasoning or simple enhancements. (2) Global RAG methods underperform on Level-1 and Level-2 tasks but excel on Level-3 and Level-4 tasks. This indicates that global reasoning is not beneficial for factoid queries and may even hinder performance. However, for rationale queries, which require synthesizing information from a broad range of text, global awareness helps gather more comprehensive evidence, leading to improved performance. (3) Directly applying long LLMs to process the entire knowledge base is feasible but underperforms on factoid queries due to over-referencing and redundant noise. However, for rationale queries, long LLMs outperform vanilla RAG methods due to their strong reasoning ability over long contexts. Efficient long-context methods, such as accelerated pre-filling or prompt compression, yield performance comparable to vanilla LLMs.

Resilience over Domains Figure 3 presents the experimental results across different levels and domains for selected methods. The results highlight how different methods perform across domain-specific knowledge: (1) For structured knowledge sources, such as financial reports and legal documents, most methods perform well on factoid queries. The inherent clarity and precision of these texts reduce semantic ambiguity, improving retrieval accuracy. (2) For explanatory texts, such as academic papers that focus on providing rationales, global RAG methods excel. Their global awareness enables them to effectively organize and integrate explicit reasoning from the knowledge base. (3) For unstructured knowledge in domains like literature, art, and humanities—where texts contain higher semantic ambiguity—global RAG methods perform better on Level-4 tasks. This suggests that aggregating high-level implicit information is more effective for narrative-based content than for structured knowledge domains.

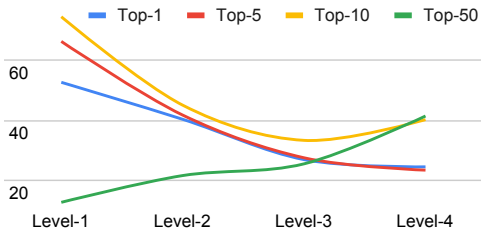

Figure 4: Evaluation performance across four levels for RAG with varying Top-k selections.

Impact of Top-k Figure 4 systematically investigates the impact of Top-k selection using vanilla RAG. The results show that while increasing Top-k introduces more knowledge into the generation

process, it also increases redundancy. The trade-off between knowledge recall and precision varies across query levels. Factoid queries rely on precise evidence, and excessive redundancy significantly degrades performance. In contrast, rationale queries benefit from higher recall, as effective information aggregation requires a more comprehensive set of evidence from the knowledge base.

Efficiency Analysis Table 2 presents a comparison of task latency across methods and task levels. The following insights can be drawn from the results:

(1) Standard RAG methods are highly efficient, as the retrieval process is not sensitive to the size of the knowledge base. In contrast, long LLMs and global RAG methods experience a notable increase in latency across all tasks, while only improving performance on rationale tasks. (2) Long LLMs incur the highest latency for all task types but fail to deliver a clear performance advantage. This suggests that directly using the full knowledge base may not be a proper approach. (3) The graph construction process for GraphRAG relies heavily on robust model APIs, leading to substantial construction latency. However, once the graph is constructed, performance becomes efficient. This indicates that optimizing the process of perceiving the global knowledge base—such as accelerating the graph construction in GraphRAG or memory formation in MemoRAG—could be beneficial for improving performance on rationale queries.

Level	RAG	HyDE	LLM	MemoRAG	GraphRAG
1	0.6	1.0	29.1	20.9	1.7 ($+\infty$)
2	0.7	2.0	32.7	21.5	2.0 ($+\infty$)
3	1.6	2.1	48.3	33.4	3.0 ($+\infty$)
4	1.7	2.2	52.1	35.9	3.5 ($+\infty$)

Table 2: Task latency (queries per second) comparison across methods and levels. Experiments were conducted on an Nvidia A800-80G GPU using the ART dataset. GraphRAG employs GPT-4o for graph construction, which can take up to half an hour, denoted by $+\infty$.

Retrieval Strategy Analysis In addition to comparing different RAG architectures, we further investigate the impact of retrieval strategies on performance across various task levels. Specifically, we evaluate three types of retrievers: **dense retrieval**, **sparse retrieval**, and a **hybrid** approach that combines both. The goal is to understand how the choice of retriever influences the resilience and adaptability of RAG systems under different information-seeking challenges.

Figure 5 presents the performance of representative RAG methods (vanilla RAG, RQRAG, and MemoRAG) using each retrieval strategy across the four task levels in HawkBench. The results demonstrate that retrieval strategy has a substantial impact on downstream performance. Dense and hybrid retrievers consistently outperform sparse retrievers, particularly on rationale-intensive tasks (Levels 3 and 4), where retrieving semantically rich information is crucial. Notably, methods that incorporate additional retrieval cues—such as query rewriting in RQRAG or memory-guided retrieval in MemoRAG—benefit significantly from hybrid retrieval. This suggests that hybrid retrievers enhance the likelihood of capturing diverse, relevant evidence when guided by auxiliary signals. These findings underscore the importance of retrieval design in RAG pipelines, especially when targeting general-purpose or reasoning-intensive tasks. Future research may explore adaptive retrieval modules that dynamically select the most suitable retrieval strategy based on task characteristics.

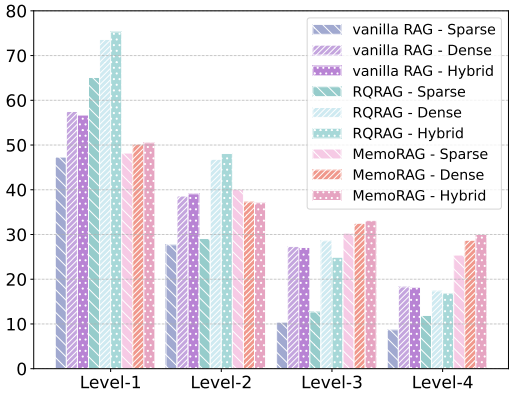

Figure 5: Performance of different retrieval strategies across the four HawkBench levels.

3.3 Key Insights

Current RAG methods Lack Resilience. Current RAG methods tend to be optimized for specific types of information-seeking tasks (e.g., fact retrieval or rationale generation). However, this specialization leads to a lack of overall resilience across a broader range of tasks. While empirical analyses provide heuristics to guide method selection for particular tasks, we still lack a systematic,

adaptable solution that can handle diverse tasks with varying requirements. This gap emphasizes the need for developing RAG systems that can dynamically adjust to different information-seeking challenges, moving beyond task-specific optimizations toward a more generalized framework.

Global Awareness: Construction and Utilization Challenges. Global awareness is essential for tasks that require the integration of information from multiple sources. However, current global RAG methods struggle with efficiently building and fully leveraging this awareness. While methods such as GraphRAG (which uses graph construction) and memory-based approaches show promise, their reliance on inefficient global intermediate construction processes (e.g., building graphs or memory stores) remains a major bottleneck. For example, graph construction can take tens of minutes, making it impractical for real-time use. Optimizing these construction processes could make these systems more viable. Additionally, there is a need for research into how to best utilize global intermediates (e.g., graphs, memory caches) to improve retrieval and reasoning. Exploring efficient ways to construct and use these intermediates is an important direction for future work.

Dynamic Task Understanding and Adaptive Query Interpretation. As information-seeking tasks become more complex, the need for dynamic task understanding and adaptive query interpretation becomes increasingly important. A one-size-fits-all solution is not feasible; instead, RAG systems must integrate decision-making mechanisms that allow them to dynamically adjust how they access (referencing) and utilize (reasoning) knowledge. By understanding the task context and adapting the retrieval strategy accordingly, RAG systems can more effectively address a wider range of queries. This adaptability would significantly enhance the robustness and efficiency of RAG methods, enabling them to handle varying complexities and task types more effectively.

The Potential of Agentic Information-Seeking Systems. Looking ahead, agentic information-seeking systems—designed to autonomously navigate knowledge acquisition—offer a compelling direction for the future of AI. By integrating retrieval, reasoning, and synthesis, these systems can perform complex tasks such as literature reviews, report writing, or exploratory research. Recent developments like OpenAI's Deep Research exemplify this trend, signaling a shift toward AI agents that not only assist but independently manage knowledge-intensive workflows. As these systems mature, they hold the potential to reshape how we interact with and generate information, making them a key area for future investigation and innovation.

4 Related Work

RAG Methods RAG was introduced by Lewis et al. [2020] to enhance language models' ability to handle knowledge-intensive tasks by providing relevant context through retrieval. Research in RAG has focused on two main areas: (1) improving retrieval quality to set an upper bound for generation accuracy [Qian et al., 2024, Gao et al., 2024], and (2) optimizing the use of retrieved passages for relevance and accessibility during generation [Jiang et al., 2023, Zhao et al., 2024b].

The integration of RAG with LLMs has gained momentum, especially in knowledge-intensive applications [Shuster et al., 2021]. As a result, there is increasing demand for more generalized RAG systems capable of handling a wider range of tasks, including those beyond factoid queries [Zhao et al., 2024a]. However, traditional RAG pipelines face challenges in addressing complex tasks with implicit information needs, often failing to provide sufficient context for accurate generation [Gao et al., 2024, Zhao et al., 2024a]. Recent advances have aimed to expand RAG's applicability. For example, *GraphRAG* [Edge et al., 2024] and *HippoRAG* [Jimenez Gutierrez et al., 2024] introduce knowledge graphs to facilitate retrieval and enhance global awareness. Agent-based approaches, such as *ActiveRAG* [Xu et al., 2024, Yoon et al., 2024], plan information access and utilization via agents.

RAG Benchmarking As RAG systems are increasingly adopted, the need for comprehensive evaluation benchmarks has become evident. Early benchmarks, such as KILT [Petroni et al., 2021], primarily focused on task-specific aspects like single-hop and multi-hop reasoning, as well as factoid queries. Recently, new benchmarks have been developed to address specialized tasks and domains. For example, MultiHop-RAG evaluates multi-hop tasks [Tang and Yang, 2024], LegalBench-RAG focuses on the legal domain [Guha et al., 2023], CRAG offers a comprehensive evaluation framework for factoid question answering tasks, and RAGBench is designed to assess the explainability of RAG systems [Friel et al., 2024]. While these benchmarks provide insights into various facets of RAG

performance, they lack a comprehensive framework to evaluate the resilience of RAG systems when faced with diverse information-seeking needs, particularly for stratified queries [Zhao et al., 2024a].

5 Conclusion

In this paper, we introduce HawkBench, a comprehensive framework designed to evaluate the resilience of RAG systems across diverse information-seeking tasks. HawkBench is distinguished by its systematic task stratification, multi-domain corpora, and high-quality annotations, making it an robust tool for assessing the resilience of RAG methods. Our evaluation of representative RAG methods reveals that while current RAG systems are often optimized for specific tasks, they lack resilience across general tasks. This highlights the need for dynamic task strategies that integrate decision-making, query interpretation, and global knowledge utilization to enhance the generalizability of RAG systems. HawkBench serves as a critical resource for advancing the development of resilient, versatile RAG systems capable of addressing a wide range of real-world user needs.

Acknowledgement

This work was supported by National Natural Science Foundation of China No. 62502049.

References

OpenAI. Gpt-4 technical report, 2023.

DeepSeek-AI. Deepseek-v3 technical report, 2024.

Yutao Zhu, Huaying Yuan, Shuting Wang, Jiongnan Liu, Wenhan Liu, Chenlong Deng, Haonan Chen, Zhicheng Dou, and Ji-Rong Wen. Large language models for information retrieval: A survey, 2024.

Hongjin Qian, Zheng Liu, Peitian Zhang, Kelong Mao, Defu Lian, Zhicheng Dou, and Tiejun Huang. Memorag: Boosting long context processing with global memory-enhanced retrieval augmentation. In *Proceedings of the ACM on Web Conference 2025*, WWW '25, page 2366–2377, New York, NY, USA, 2025. Association for Computing Machinery.

Siyun Zhao, Yuqing Yang, Zilong Wang, Zhiyuan He, Luna K. Qiu, and Lili Qiu. Retrieval augmented generation (rag) and beyond: A comprehensive survey on how to make your llms use external data more wisely, 2024a.

Qingfei Zhao, Ruobing Wang, Yukuo Cen, Daren Zha, Shicheng Tan, Yuxiao Dong, and Jie Tang. Longrag: A dual-perspective retrieval-augmented generation paradigm for long-context question answering. In Yaser Al-Onaizan, Mohit Bansal, and Yun-Nung Chen, editors, *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, EMNLP 2024, Miami, FL, USA, November 12-16, 2024*, pages 22600–22632. Association for Computational Linguistics, 2024b.

Zhipeng Xu, Zhenghao Liu, Yibin Liu, Chenyan Xiong, Yukun Yan, Shuo Wang, Shi Yu, Zhiyuan Liu, and Ge Yu. Activerag: Revealing the treasures of knowledge via active learning. *CoRR*, abs/2402.13547, 2024.

Darren Edge, Ha Trinh, Newman Cheng, Joshua Bradley, Alex Chao, Apurva Mody, Steven Truitt, and Jonathan Larson. From local to global: A graph rag approach to query-focused summarization, 2024.

Neel Guha, Julian Nyarko, Daniel Ho, Christopher Ré, Adam Chilton, Alex Chohlas-Wood, Austin Peters, Brandon Waldon, Daniel Rockmore, Diego Zambrano, et al. Legalbench: A collaboratively built benchmark for measuring legal reasoning in large language models. *Advances in neural information processing systems*, 36:44123–44279, 2023.

Yixuan Tang and Yi Yang. Multihop-rag: Benchmarking retrieval-augmented generation for multi-hop queries. In *First Conference on Language Modeling*, 2024.

Xiao Yang, Kai Sun, Hao Xin, Yushi Sun, Nikita Bhalla, Xiangsen Chen, Sajal Choudhary, Rongze Daniel Gui, Ziran Will Jiang, Ziyu Jiang, Lingkun Kong, Brian Moran, Jiaqi Wang, Yifan Xu, An Yan, Chenyu Yang, Eting Yuan, Hanwen Zha, Nan Tang, Lei Chen, Nicolas Scheffer, Yue Liu, Nirav Shah, Rakesh Wanga, Anuj Kumar, Scott Yih, and Xin Dong. CRAG - comprehensive RAG benchmark. In Amir Globersons, Lester Mackey, Danielle Belgrave, Angela Fan, Ulrich Paquet, Jakub M. Tomczak, and Cheng Zhang, editors, *Advances in Neural Information Processing Systems 38: Annual Conference on Neural Information Processing Systems 2024, NeurIPS 2024, Vancouver, BC, Canada, December 10 - 15, 2024*, 2024.

Luyu Gao, Xueguang Ma, Jimmy Lin, and Jamie Callan. Precise zero-shot dense retrieval without relevance labels. In Anna Rogers, Jordan L. Boyd-Graber, and Naoaki Okazaki, editors, *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 1762–1777. Association for Computational Linguistics, 2023.

Chi-Min Chan, Chunpu Xu, Ruibin Yuan, Hongyin Luo, Wei Xue, Yike Guo, and Jie Fu. Rq-rag: Learning to refine queries for retrieval augmented generation. In *First Conference on Language Modeling*, 2024.

Zhuoshi Pan, Qianhui Wu, Huiqiang Jiang, Menglin Xia, Xufang Luo, Jue Zhang, Qingwei Lin, Victor Rühle, Yuqing Yang, Chin-Yew Lin, et al. Llmlingua-2: Data distillation for efficient and faithful task-agnostic prompt compression. In *ACL (Findings)*, 2024.

Huiqiang Jiang, Yucheng Li, Chengruidong Zhang, Qianhui Wu, Xufang Luo, Surin Ahn, Zhenhua Han, Amir H Abdi, Dongsheng Li, Chin-Yew Lin, et al. Minference 1.0: Accelerating pre-filling for long-context llms via dynamic sparse attention. *Advances in Neural Information Processing Systems*, 37:52481–52515, 2024.

Qwen Team. Qwen2.5 technical report, 2025.

Jianlv Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. Bge m3-embedding: Multi-lingual, multi-functionality, multi-granularity text embeddings through self-knowledge distillation, 2023.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-Augmented Generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474, 2020.

Hongjin Qian, Zheng Liu, Kelong Mao, Yujia Zhou, and Zhicheng Dou. Grounding language model with chunking-free in-context retrieval. In Lun-Wei Ku, Andre Martins, and Vivek Srikumar, editors, *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 1298–1311. Association for Computational Linguistics, 2024.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Qianyu Guo, Meng Wang, and Haofen Wang. Retrieval-augmented generation for large language models: A survey, 2024.

Zhengbao Jiang, Frank F. Xu, Luyu Gao, Zhiqing Sun, Qian Liu, Jane Dwivedi-Yu, Yiming Yang, Jamie Callan, and Graham Neubig. Active retrieval augmented generation. In Houda Bouamor, Juan Pino, and Kalika Bali, editors, *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 7969–7992. Association for Computational Linguistics, 2023.

Kurt Shuster, Spencer Poff, Moya Chen, Douwe Kiela, and Jason Weston. Retrieval augmentation reduces hallucination in conversation. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih, editors, *Findings of the Association for Computational Linguistics: EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 16-20 November, 2021*, pages 3784–3803. Association for Computational Linguistics, 2021.

Bernal Jimenez Gutierrez, Yiheng Shu, Yu Gu, Michihiro Yasunaga, and Yu Su. Hipporag: Neurobiologically inspired long-term memory for large language models. *Advances in Neural Information Processing Systems*, 37:59532–59569, 2024.

Chanwoong Yoon, Taewhoo Lee, Hyeon Hwang, Minbyul Jeong, and Jaewoo Kang. Compact: Compressing retrieved documents actively for question answering. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 21424–21439, 2024.

Fabio Petroni, Aleksandra Piktus, Angela Fan, Patrick S. H. Lewis, Majid Yazdani, Nicola De Cao, James Thorne, Yacine Jernite, Vladimir Karpukhin, Jean Maillard, Vassilis Plachouras, Tim Rocktäschel, and Sebastian Riedel. KILT: a benchmark for knowledge intensive language tasks. In Kristina Toutanova, Anna Rumshisky, Luke Zettlemoyer, Dilek Hakkani-Tür, Iz Beltagy, Steven Bethard, Ryan Cotterell, Tanmoy Chakraborty, and Yichao Zhou, editors, *NAACL-HLT 2021, Online, June 6-11, 2021*, pages 2523–2544. Association for Computational Linguistics, 2021.

Robert Friel, Masha Belyi, and Atindriyo Sanyal. Ragbench: Explainable benchmark for retrieval-augmented generation systems. *CoRR*, abs/2407.11005, 2024.

NeurIPS Paper Checklist

1. **Claims**

   Question: Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope?

   Answer: [Yes]

   Justification: In the experiment section, including the main experiments and discussions.

   Guidelines:

   - The answer NA means that the abstract and introduction do not include the claims made in the paper.
   - The abstract and/or introduction should clearly state the claims made, including the contributions made in the paper and important assumptions and limitations. A No or NA answer to this question will not be perceived well by the reviewers.
   - The claims made should match theoretical and experimental results, and reflect how much the results can be expected to generalize to other settings.
   - It is fine to include aspirational goals as motivation as long as it is clear that these goals are not attained by the paper.

2. **Limitations**

   Question: Does the paper discuss the limitations of the work performed by the authors?

   Answer:[Yes]

   Justification: In Appendix, we have a Limitation section.

   Guidelines:

   - The answer NA means that the paper has no limitation while the answer No means that the paper has limitations, but those are not discussed in the paper.
   - The authors are encouraged to create a separate "Limitations" section in their paper.
   - The paper should point out any strong assumptions and how robust the results are to violations of these assumptions (e.g., independence assumptions, noiseless settings, model well-specification, asymptotic approximations only holding locally). The authors should reflect on how these assumptions might be violated in practice and what the implications would be.
   - The authors should reflect on the scope of the claims made, e.g., if the approach was only tested on a few datasets or with a few runs. In general, empirical results often depend on implicit assumptions, which should be articulated.
   - The authors should reflect on the factors that influence the performance of the approach. For example, a facial recognition algorithm may perform poorly when image resolution is low or images are taken in low lighting. Or a speech-to-text system might not be used reliably to provide closed captions for online lectures because it fails to handle technical jargon.
   - The authors should discuss the computational efficiency of the proposed algorithms and how they scale with dataset size.
   - If applicable, the authors should discuss possible limitations of their approach to address problems of privacy and fairness.
   - While the authors might fear that complete honesty about limitations might be used by reviewers as grounds for rejection, a worse outcome might be that reviewers discover limitations that aren't acknowledged in the paper. The authors should use their best judgment and recognize that individual actions in favor of transparency play an important role in developing norms that preserve the integrity of the community. Reviewers will be specifically instructed to not penalize honesty concerning limitations.

3. **Theory assumptions and proofs**

   Question: For each theoretical result, does the paper provide the full set of assumptions and a complete (and correct) proof?

   Answer: [NA]

Justification: This paper does not contain theoretical results.

Guidelines:

- The answer NA means that the paper does not include theoretical results.
- All the theorems, formulas, and proofs in the paper should be numbered and cross-referenced.
- All assumptions should be clearly stated or referenced in the statement of any theorems.
- The proofs can either appear in the main paper or the supplemental material, but if they appear in the supplemental material, the authors are encouraged to provide a short proof sketch to provide intuition.
- Inversely, any informal proof provided in the core of the paper should be complemented by formal proofs provided in appendix or supplemental material.
- Theorems and Lemmas that the proof relies upon should be properly referenced.

4. **Experimental result reproducibility**

   Question: Does the paper fully disclose all the information needed to reproduce the main experimental results of the paper to the extent that it affects the main claims and/or conclusions of the paper (regardless of whether the code and data are provided or not)?

   Answer:[Yes]

   Justification: We have disclosed necessary details for our result reproducibility.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - If the paper includes experiments, a No answer to this question will not be perceived well by the reviewers: Making the paper reproducible is important, regardless of whether the code and data are provided or not.
   - If the contribution is a dataset and/or model, the authors should describe the steps taken to make their results reproducible or verifiable.
   - Depending on the contribution, reproducibility can be accomplished in various ways. For example, if the contribution is a novel architecture, describing the architecture fully might suffice, or if the contribution is a specific model and empirical evaluation, it may be necessary to either make it possible for others to replicate the model with the same dataset, or provide access to the model. In general. releasing code and data is often one good way to accomplish this, but reproducibility can also be provided via detailed instructions for how to replicate the results, access to a hosted model (e.g., in the case of a large language model), releasing of a model checkpoint, or other means that are appropriate to the research performed.
   - While NeurIPS does not require releasing code, the conference does require all submissions to provide some reasonable avenue for reproducibility, which may depend on the nature of the contribution. For example
     (a) If the contribution is primarily a new algorithm, the paper should make it clear how to reproduce that algorithm.
     (b) If the contribution is primarily a new model architecture, the paper should describe the architecture clearly and fully.
     (c) If the contribution is a new model (e.g., a large language model), then there should either be a way to access this model for reproducing the results or a way to reproduce the model (e.g., with an open-source dataset or instructions for how to construct the dataset).
     (d) We recognize that reproducibility may be tricky in some cases, in which case authors are welcome to describe the particular way they provide for reproducibility. In the case of closed-source models, it may be that access to the model is limited in some way (e.g., to registered users), but it should be possible for other researchers to have some path to reproducing or verifying the results.

5. **Open access to data and code**

   Question: Does the paper provide open access to the data and code, with sufficient instructions to faithfully reproduce the main experimental results, as described in supplemental material?

Answer:[Yes]

Justification: In Appendix, we have a section to discuss the implementation details of our paper. We also provide source codes and training script in the supplement material.

Guidelines:

- The answer NA means that paper does not include experiments requiring code.
- Please see the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- While we encourage the release of code and data, we understand that this might not be possible, so "No" is an acceptable answer. Papers cannot be rejected simply for not including code, unless this is central to the contribution (e.g., for a new open-source benchmark).
- The instructions should contain the exact command and environment needed to run to reproduce the results. See the NeurIPS code and data submission guidelines (`https://nips.cc/public/guides/CodeSubmissionPolicy`) for more details.
- The authors should provide instructions on data access and preparation, including how to access the raw data, preprocessed data, intermediate data, and generated data, etc.
- The authors should provide scripts to reproduce all experimental results for the new proposed method and baselines. If only a subset of experiments are reproducible, they should state which ones are omitted from the script and why.
- At submission time, to preserve anonymity, the authors should release anonymized versions (if applicable).
- Providing as much information as possible in supplemental material (appended to the paper) is recommended, but including URLs to data and code is permitted.

6. **Experimental setting/details**

   Question: Does the paper specify all the training and test details (e.g., data splits, hyper-parameters, how they were chosen, type of optimizer, etc.) necessary to understand the results?

   Answer:[Yes]

   Justification: In Appendix, we have a implementaion details section to disclose these details.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The experimental setting should be presented in the core of the paper to a level of detail that is necessary to appreciate the results and make sense of them.
   - The full details can be provided either with the code, in appendix, or as supplemental material.

7. **Experiment statistical significance**

   Question: Does the paper report error bars suitably and correctly defined or other appropriate information about the statistical significance of the experiments?

   Answer:[Yes]

   Justification: We did t-test for the main experiments.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The authors should answer "Yes" if the results are accompanied by error bars, confidence intervals, or statistical significance tests, at least for the experiments that support the main claims of the paper.
   - The factors of variability that the error bars are capturing should be clearly stated (for example, train/test split, initialization, random drawing of some parameter, or overall run with given experimental conditions).
   - The method for calculating the error bars should be explained (closed form formula, call to a library function, bootstrap, etc.)
   - The assumptions made should be given (e.g., Normally distributed errors).

- It should be clear whether the error bar is the standard deviation or the standard error of the mean.
- It is OK to report 1-sigma error bars, but one should state it. The authors should preferably report a 2-sigma error bar than state that they have a 96% CI, if the hypothesis of Normality of errors is not verified.
- For asymmetric distributions, the authors should be careful not to show in tables or figures symmetric error bars that would yield results that are out of range (e.g. negative error rates).
- If error bars are reported in tables or plots, The authors should explain in the text how they were calculated and reference the corresponding figures or tables in the text.

8. **Experiments compute resources**

   Question: For each experiment, does the paper provide sufficient information on the computer resources (type of compute workers, memory, time of execution) needed to reproduce the experiments?

   Answer:[Yes]

   Justification: We disclose the required computing resources in the implementation details section.

   Guidelines:

   - The answer NA means that the paper does not include experiments.
   - The paper should indicate the type of compute workers CPU or GPU, internal cluster, or cloud provider, including relevant memory and storage.
   - The paper should provide the amount of compute required for each of the individual experimental runs as well as estimate the total compute.
   - The paper should disclose whether the full research project required more compute than the experiments reported in the paper (e.g., preliminary or failed experiments that didn't make it into the paper).

9. **Code of ethics**

   Question: Does the research conducted in the paper conform, in every respect, with the NeurIPS Code of Ethics `https://neurips.cc/public/EthicsGuidelines`?

   Answer:[Yes]

   Justification: We follow the NeurIPS Code of Ethics.

   Guidelines:

   - The answer NA means that the authors have not reviewed the NeurIPS Code of Ethics.
   - If the authors answer No, they should explain the special circumstances that require a deviation from the Code of Ethics.
   - The authors should make sure to preserve anonymity (e.g., if there is a special consideration due to laws or regulations in their jurisdiction).

10. **Broader impacts**

    Question: Does the paper discuss both potential positive societal impacts and negative societal impacts of the work performed?

    Answer:[Yes]

    Justification: We discuss the impact of this paper in Appendix.

    Guidelines:

    - The answer NA means that there is no societal impact of the work performed.
    - If the authors answer NA or No, they should explain why their work has no societal impact or why the paper does not address societal impact.
    - Examples of negative societal impacts include potential malicious or unintended uses (e.g., disinformation, generating fake profiles, surveillance), fairness considerations (e.g., deployment of technologies that could make decisions that unfairly impact specific groups), privacy considerations, and security considerations.

- The conference expects that many papers will be foundational research and not tied to particular applications, let alone deployments. However, if there is a direct path to any negative applications, the authors should point it out. For example, it is legitimate to point out that an improvement in the quality of generative models could be used to generate deepfakes for disinformation. On the other hand, it is not needed to point out that a generic algorithm for optimizing neural networks could enable people to train models that generate Deepfakes faster.
- The authors should consider possible harms that could arise when the technology is being used as intended and functioning correctly, harms that could arise when the technology is being used as intended but gives incorrect results, and harms following from (intentional or unintentional) misuse of the technology.
- If there are negative societal impacts, the authors could also discuss possible mitigation strategies (e.g., gated release of models, providing defenses in addition to attacks, mechanisms for monitoring misuse, mechanisms to monitor how a system learns from feedback over time, improving the efficiency and accessibility of ML).

11. **Safeguards**

    Question: Does the paper describe safeguards that have been put in place for responsible release of data or models that have a high risk for misuse (e.g., pretrained language models, image generators, or scraped datasets)?

    Answer:[NA]

    Justification: the models in this paper are trained for specific search scenarios, which does not pose such risks.

    Guidelines:

    - The answer NA means that the paper poses no such risks.
    - Released models that have a high risk for misuse or dual-use should be released with necessary safeguards to allow for controlled use of the model, for example by requiring that users adhere to usage guidelines or restrictions to access the model or implementing safety filters.
    - Datasets that have been scraped from the Internet could pose safety risks. The authors should describe how they avoided releasing unsafe images.
    - We recognize that providing effective safeguards is challenging, and many papers do not require this, but we encourage authors to take this into account and make a best faith effort.

12. **Licenses for existing assets**

    Question: Are the creators or original owners of assets (e.g., code, data, models), used in the paper, properly credited and are the license and terms of use explicitly mentioned and properly respected?

    Answer:[Yes]

    Justification: We credited all used resources in this paper.

    Guidelines:

    - The answer NA means that the paper does not use existing assets.
    - The authors should cite the original paper that produced the code package or dataset.
    - The authors should state which version of the asset is used and, if possible, include a URL.
    - The name of the license (e.g., CC-BY 4.0) should be included for each asset.
    - For scraped data from a particular source (e.g., website), the copyright and terms of service of that source should be provided.
    - If assets are released, the license, copyright information, and terms of use in the package should be provided. For popular datasets, `paperswithcode.com/datasets` has curated licenses for some datasets. Their licensing guide can help determine the license of a dataset.
    - For existing datasets that are re-packaged, both the original license and the license of the derived asset (if it has changed) should be provided.

- If this information is not available online, the authors are encouraged to reach out to the asset's creators.

13. **New assets**

    Question: Are new assets introduced in the paper well documented and is the documentation provided alongside the assets?

    Answer:[Yes]

    Justification: We introduced the details about our constructed training data in the main content.

    Guidelines:

    - The answer NA means that the paper does not release new assets.
    - Researchers should communicate the details of the dataset/code/model as part of their submissions via structured templates. This includes details about training, license, limitations, etc.
    - The paper should discuss whether and how consent was obtained from people whose asset is used.
    - At submission time, remember to anonymize your assets (if applicable). You can either create an anonymized URL or include an anonymized zip file.

14. **Crowdsourcing and research with human subjects**

    Question: For crowdsourcing experiments and research with human subjects, does the paper include the full text of instructions given to participants and screenshots, if applicable, as well as details about compensation (if any)?

    Answer:[Yes]

    Justification: In the Appendix, we have screenshots of the annotation system. In the supplementary materials, we provide codes for the annotation system.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Including this information in the supplemental material is fine, but if the main contribution of the paper involves human subjects, then as much detail as possible should be included in the main paper.
    - According to the NeurIPS Code of Ethics, workers involved in data collection, curation, or other labor should be paid at least the minimum wage in the country of the data collector.

15. **Institutional review board (IRB) approvals or equivalent for research with human subjects**

    Question: Does the paper describe potential risks incurred by study participants, whether such risks were disclosed to the subjects, and whether Institutional Review Board (IRB) approvals (or an equivalent approval/review based on the requirements of your country or institution) were obtained?

    Answer:[NA]

    Justification: The paper does not involve crowdsourcing nor research with human subjects.

    Guidelines:

    - The answer NA means that the paper does not involve crowdsourcing nor research with human subjects.
    - Depending on the country in which research is conducted, IRB approval (or equivalent) may be required for any human subjects research. If you obtained IRB approval, you should clearly state this in the paper.
    - We recognize that the procedures for this may vary significantly between institutions and locations, and we expect authors to adhere to the NeurIPS Code of Ethics and the guidelines for their institution.
    - For initial submissions, do not include any information that would break anonymity (if applicable), such as the institution conducting the review.

16. **Declaration of LLM usage**

Question: Does the paper describe the usage of LLMs if it is an important, original, or non-standard component of the core methods in this research? Note that if the LLM is used only for writing, editing, or formatting purposes and does not impact the core methodology, scientific rigorousness, or originality of the research, declaration is not required.

Answer:[Yes]

Justification: We have described the usage of LLMs as a core component of our method in the paper.

Guidelines:

- The answer NA means that the core method development in this research does not involve LLMs as any important, original, or non-standard components.
- Please refer to our LLM policy (`https://neurips.cc/Conferences/2025/LLM`) for what should or should not be described.

A Implementation Details

In our evaluation of baseline methods on HawkRAG, we use BGE-M3 [Chen et al., 2023] as the retriever for vanilla RAG, RQ-RAG, HyDE, and MemoRAG, setting the hit number to 5. For methods that segment long contexts into chunks, we utilize the semantic-text-splitter tool, limiting chunks to a maximum of 512 tokens. MemoRAG employs the officially released memorag-qwen2-7b-inst as its memory model. For GraphRAG, we leverage GPT-4o for graph construction and use the retrieved context for generation. All baseline methods adopt Qwen-2.5-7B-instruct-128K as the generator.

HawkRAG's raw texts are sourced from books-3-textbooks, legal contracts, arXiv papers, and financial reports. During annotation, the annotator would select either GPT-4o or DeepSeek-v3 as the assisting agent. Our annotation system, illustrated in Figure 6, is implemented using Streamlit. The statistic details of HawkBench are presented in Table 3. In Table 5, we present the full results of the main experiments.

All experiments were conducted on a server equipped with 8 NVIDIA A800-80G GPUs.

Table 3: Statistical Information of HawkBench. The symbols $\langle|\mathcal{C}|\rangle$, $\langle|\mathcal{Q}|\rangle$, and $\langle|\mathcal{A}|\rangle$ represent the average lengths of the context, query, and answer, respectively.

Dataset	Num	$\langle\|\mathcal{C}\|\rangle$	Num	$\langle\|\mathcal{Q}\|\rangle$	$\langle\|\mathcal{A}\|\rangle$	Num	$\langle\|\mathcal{Q}\|\rangle$	$\langle\|\mathcal{A}\|\rangle$	Num	$\langle\|\mathcal{Q}\|\rangle$	$\langle\|\mathcal{A}\|\rangle$	Num	$\langle\|\mathcal{Q}\|\rangle$	$\langle\|\mathcal{A}\|\rangle$
			LEVEL-1			LEVEL-2			LEVEL-3			LEVEL-4		
TECHNOLOGY	200	144803.0	50	15.8	5.1	50	57.7	14.1	50	25.3	96.4	50	26.3	42.0
NOVEL	200	166960.2	50	14.2	6.8	50	51.6	19.0	50	28.2	121.5	50	31.1	63.5
ART	200	115591.8	50	17.0	6.9	50	53.6	14.8	50	27.0	125.2	50	34.4	87.7
HUMANITIES	200	152600.3	50	16.8	6.9	50	56.1	26.6	50	29.1	134.1	50	33.6	72.3
PAPER	200	41702.0	50	18.2	9.5	50	75.7	17.1	50	34.0	101.0	50	28.6	40.3
SCIENCE	200	143517.0	50	16.3	7.6	50	54.3	15.3	50	26.8	109.2	50	29.0	47.9
FINANCE	200	37364.6	50	17.2	10.5	50	62.6	12.5	50	27.0	105.6	50	28.0	65.0
LEGAL	200	49331.1	50	19.3	11.9	50	53.0	21.0	50	27.2	113.0	50	27.0	46.7
Total	1600	106483.7	400	16.8	8.2	400	58.1	17.5	400	28.1	113.3	400	29.7	58.2

B Limitations

This paper focuses on constructing a benchmark, HawkBench, to evaluate the resilience of RAG methods across stratified tasks. While the benchmark provides a comprehensive framework, there are several limitations to consider. First, dataset bias may arise during the curation process, as the raw data are collected from multiple domains. This diversity, while beneficial, may inadvertently introduce biases that could affect the generalizability of the results. Additionally, during the annotation process, both the assisting LLMs and human annotators may introduce errors, which could impact the overall evaluation quality. Although we strive for thoroughness in evaluating task and domain diversity, HawkBench's size, while reasonable, may not cover all professional knowledge-intensive domains or task types.

Furthermore, while we conduct comprehensive experiments using HawkBench, it is not feasible to test all available RAG methods, alternative retrievers, or LLMs on this benchmark. We selected representative methods and models that are expected to provide generalizable findings, but this selection does not encompass the full range of possible approaches. Additionally, we did not evaluate commercial RAG solutions in this study, as these systems are typically closed-sourced and subject to changes over time, making them challenging to incorporate into a static benchmark evaluation.

C Public Data and Model Memorization vs. Genuine Retrieval

Most publicly available web data, including domain-specific corpora, are likely included in the pre-training corpus of today's large language models. This challenge is shared by nearly all modern NLP benchmarks.

Nevertheless, benchmarks built on such corpora remain meaningful for several reasons. First, seeing a text during pre-training does not guarantee full memorization, nor does it ensure accurate answers for queries requiring complex reasoning or synthesis. Our benchmark's query–answer pairs are manually

Domain & Level	gemini-2.5-flash	gpt-4o-mini	gpt-4.1	Best RAG (Qwen-2.5 7B)
Legal-Level 1	13.6	10.5	15.3	**79.2**
Legal-Level 2	25.2	20.6	30.0	**45.9**
Legal-Level 3	15.8	20.7	22.0	**32.7**
Legal-Level 4	12.4	13.2	14.3	**17.9**
Finance-Level 1	13.6	12.3	22.3	**79.5**
Finance-Level 2	25.2	13.3	23.1	**54.7**
Finance-Level 3	13.8	17.8	20.2	**30.8**
Finance-Level 4	13.4	15.8	18.9	**20.4**
Science-Level 1	13.6	12.2	13.2	**45.1**
Science-Level 2	25.1	17.8	21.2	**33.8**
Science-Level 3	15.7	21.4	20.1	**27.3**
Science-Level 4	13.4	17.2	17.4	**20.4**

Table 4: Comparison of commercial LLM APIs with the best-performing RAG system (Qwen-2.5 7B). RAG substantially outperforms LLMs in factoid queries (Level 1 and 2), while LLMs remain competitive in higher-level reasoning tasks (Level 3 and 4).

annotated to capture nuanced, multi-step information-seeking behaviors that go well beyond simple fact recall.

Second, to mitigate concerns regarding memorization versus genuine retrieval, we include evaluations using several strong commercial LLM APIs. By comparing the performance of a range of models on our benchmark, we can better assess the extent to which retrieval-augmented reasoning (rather than memorization) contributes to success. Specifically, we compare RAG methods with three leading commercial LLMs. The results are shown in Table 4.

The results demonstrate that while strong LLMs have memorized substantial information from public corpora, they still lag behind retrieval-augmented methods in overall performance. Notably, for factoid queries at Level 1 and Level 2, RAG methods outperform strong LLMs by a large margin, suggesting that even with exposure to the underlying texts during pre-training, LLMs cannot reliably recall fine-grained factual details. For Level 3 and Level 4 tasks, which require summarizing broad content or synthesizing information, strong LLMs perform comparably to RAG methods, as these queries demand less precise retrieval and more general reasoning.

In summary, these results show that **even though portions of HawkBench may have been seen by strong LLMs during pretraining, it remains a robust benchmark for evaluating stratified RAG performance**. Without retrieval, even advanced LLMs such as GPT-4.1 can only solve a small fraction of the tasks, highlighting the necessity of effective retrieval-augmented reasoning. Moreover, these experiments suggest that HawkBench not only provides a comprehensive testbed for RAG evaluation, but also serves as a tool for assessing the factual memorization capabilities of state-of-the-art LLMs.

D Broader Impact

Our work aims to advance the robustness and generalizability of RAG systems by introducing a comprehensive benchmark, HawkBench, that stratifies tasks based on real-world information-seeking complexity. This can benefit a wide range of applications—such as question answering, legal and financial document analysis, and educational tutoring—by enabling more adaptive and reliable retrieval-augmented language models.

However, improving general-purpose information-seeking systems also raises concerns. These include the risk of propagating misinformation from retrieved content, amplifying biases present in training or retrieval corpora, and enabling misuse in sensitive domains without sufficient oversight. We encourage developers to adopt careful evaluation and safeguards when deploying RAG systems, especially in high-stakes or regulated scenarios.

Ultimately, we hope that HawkBench facilitates more transparent, equitable, and effective development of retrieval-based AI systems, while fostering research into more accountable and context-aware reasoning mechanisms.

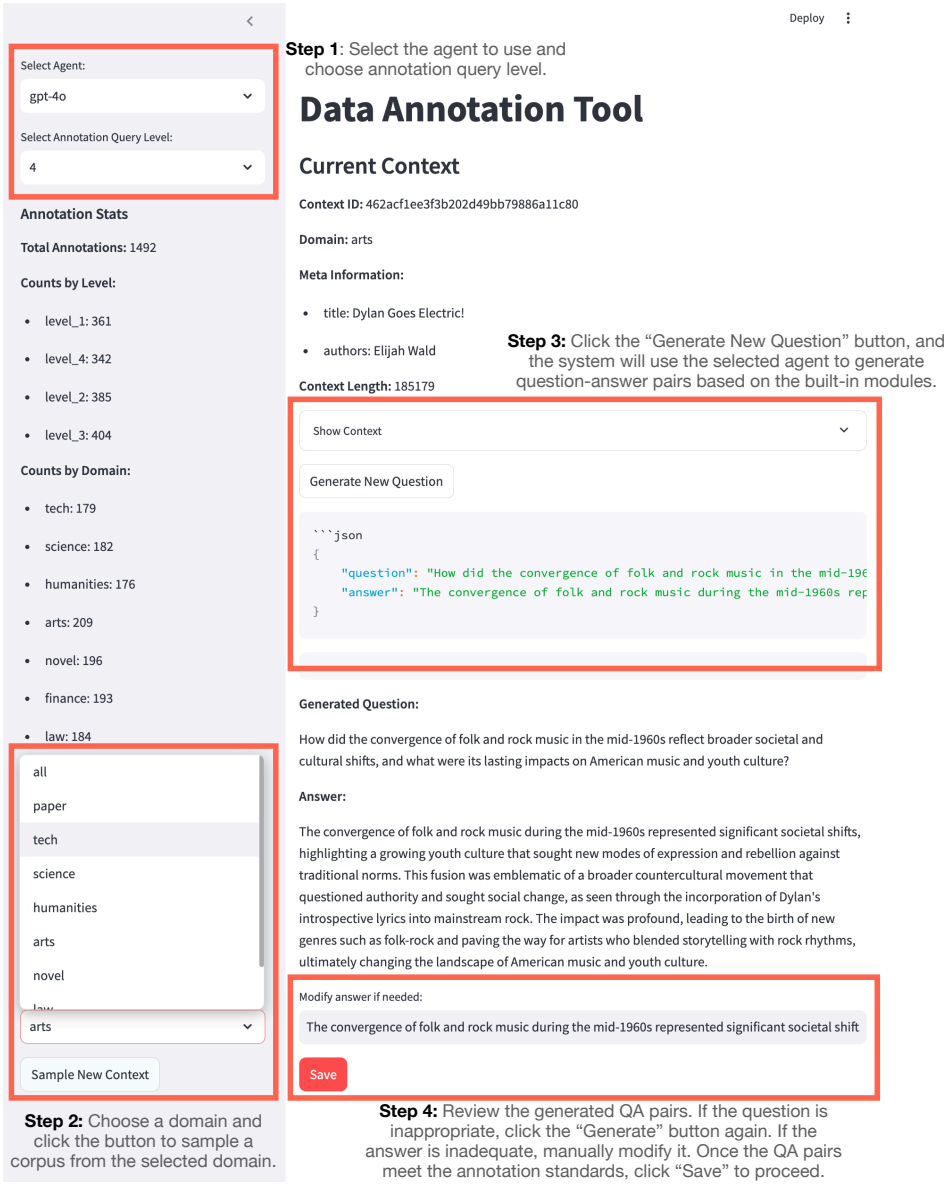

Figure 6: Annotation Interface of HawkBench.

Table 5: Full details of main experimental results.

Dataset	TECH		NOV		ART		HUM		PAPER		SCI		FIN		LEG		AVE	
LEVEL-1	R-L	F1	R-L	F1	R-L	F1	R-L	F1	R-L	F1	R-L	F1	R-L	F1	R-L	F1	R-L	F1
Top-1	47.8	52.6	39.9	42.4	39.5	46.2	25.2	27.9	30.8	32.7	29.3	44.1	87.2	88.5	38.5	39.1	42.3	46.7
Top-5	59.0	66.2	47.0	55.1	44.2	54.0	23.5	28.8	58.2	64.0	44.6	57.1	80.3	83.9	50.4	50.9	50.9	57.5
Top-10	69.0	74.4	58.3	70.7	56.8	72.2	58.6	69.4	57.5	62.1	45.6	65.4	77.6	80.9	74.1	75.0	62.2	71.3
Top-50	12.8	12.7	33.8	35.9	25.3	29.3	14.1	16.4	27.6	28.0	20.4	23.3	43.5	44.2	35.3	37.1	26.6	28.4
LLM	7.5	6.8	6.3	6.1	7.1	6.9	6.8	5.8	20.1	20.4	8.2	9.1	24.6	24.8	23.5	23.3	13.0	12.9
Lingua-2	5.0	3.5	3.8	3.0	8.9	9.1	3.0	2.1	8.2	7.4	5.9	6.2	27.0	28.2	29.8	31.4	11.4	11.4
MInference	8.0	7.1	5.3	5.1	7.2	7.1	5.3	5.1	15.7	13.9	7.2	7.7	23.3	23.2	19.7	19.7	11.5	11.1
HyDE	71.2	78.2	57.2	69.4	56.5	71.4	62.5	72.9	63.6	66.7	47.3	67.5	79.5	82.1	77.8	79.5	64.4	73.5
RQRAG	72.0	78.2	55.4	68.3	59.0	74.2	57.5	70.7	65.7	67.6	45.1	66.4	80.9	83.9	78.2	79.2	64.2	73.6
MemoRAG	46.9	51.4	29.6	34.9	35.1	46.0	48.5	55.0	32.9	35.1	35.1	44.6	65.4	68.2	65.1	66.5	44.8	50.2
GraphRAG	58.7	66.8	52.5	58.0	44.3	55.9	48.1	57.0	28.8	34.9	39.6	58.5	67.7	72.7	54.3	55.3	49.3	57.4
Dataset	TECH		NOV		ART		HUM		PAPER		SCI		FIN		LEG		AVE	
LEVEL-2	R-L	F1	R-L	F1	R-L	F1	R-L	F1	R-L	F1	R-L	F1	R-L	F1	R-L	F1	R-L	F1
Top-1	35.6	40.3	20.1	22.5	27.4	33.4	26.7	32.2	32.8	38.0	22.3	25.5	40.5	42.5	33.1	37.1	29.8	34.0
Top-5	37.1	41.7	28.9	35.8	29.8	34.8	30.8	36.6	40.9	45.6	23.0	25.1	44.0	47.1	37.1	41.7	34.0	38.6
Top-10	39.1	45.0	40.9	49.3	31.9	40.4	35.0	39.6	40.9	43.9	28.9	31.7	51.9	54.2	45.4	51.5	39.3	44.4
Top-50	20.1	21.6	25.2	26.8	18.9	21.3	22.7	25.4	21.0	19.2	16.8	18.4	23.0	22.0	25.3	27.5	21.6	22.8
LLM	10.1	8.6	11.4	10.6	11.5	11.7	14.8	12.8	17.1	14.2	9.8	8.8	10.4	8.5	18.4	16.6	12.9	11.5
Lingua-2	10.2	8.9	10.1	9.3	8.9	9.8	13.2	12.6	14.6	12.2	9.8	8.9	13.2	12.1	17.5	17.5	12.2	11.4
MInference	9.7	8.1	11.8	11.6	11.0	10.7	13.9	12.3	15.9	13.6	9.9	8.7	10.7	8.8	17.6	15.7	12.6	11.2
HyDE	45.3	48.6	39.7	46.7	33.5	39.3	33.2	40.2	39.0	43.4	30.2	32.3	54.3	55.6	45.1	49.9	40.0	44.5
RQRAG	44.4	48.0	38.5	46.2	36.5	45.4	33.3	40.9	41.5	47.5	33.8	37.2	54.7	58.6	45.9	50.2	41.1	46.8
MemoRAG	33.0	37.9	26.2	30.4	30.2	36.0	31.1	35.3	38.8	42.0	24.7	26.1	46.6	48.1	39.2	42.6	33.7	37.3
GraphRAG	34.8	38.9	35.9	40.7	28.9	30.9	33.5	38.7	31.7	31.0	25.0	27.4	45.9	48.6	36.5	39.4	34.0	37.0
Dataset	TECH		NOV		ART		HUM		PAPER		SCI		FIN		LEG		AVE	
LEVEL-3	R-L	$\mathcal{S}$-F1	R-L	$\mathcal{S}$-F1	R-L	$\mathcal{S}$-F1	R-L	$\mathcal{S}$-F1	R-L	$\mathcal{S}$-F1	R-L	$\mathcal{S}$-F1	R-L	$\mathcal{S}$-F1	R-L	$\mathcal{S}$-F1	R-L	$\mathcal{S}$-F1
Top-1	15.5	26.9	12.4	14.6	12.3	26.0	10.1	19.9	20.3	23.5	17.8	26.0	12.6	13.7	19.2	23.2	15.0	21.7
Top-5	15.5	27.5	15.7	22.2	14.4	30.0	16.4	22.1	22.9	27.7	19.0	34.9	17.2	22.5	22.3	31.4	17.9	27.3
Top-10	22.3	33.3	20.4	22.8	18.4	35.6	19.2	28.1	30.3	44.9	24.2	43.6	22.4	26.3	27.9	33.1	23.1	33.5
Top-50	18.9	25.5	19.8	23.3	19.3	30.5	24.5	29.7	26.5	30.7	23.9	37.2	27.1	31.2	26.7	35.4	23.3	30.4
LLM	23.8	20.1	23.3	17.4	23.2	19.2	23.7	23.8	30.3	34.8	24.6	26.1	30.2	30.9	30.3	26.5	26.2	24.9
Lingua-2	19.8	16.3	21.9	17.4	19.9	18.4	20.6	18.5	26.7	31.8	22.0	19.5	30.8	35.9	28.4	33.9	23.7	23.9
MInference	23.3	19.8	23.2	16.8	22.1	20.2	23.5	23.8	29.9	34.8	24.6	25.8	29.4	25.0	28.8	27.3	25.6	24.2
HyDE	17.7	34.6	16.0	14.1	16.7	33.5	16.4	21.6	26.1	35.7	21.6	36.7	16.6	24.3	24.1	23.5	19.4	28.0
RQRAG	17.6	31.7	15.7	23.1	17.8	32.4	16.3	20.9	25.3	32.9	20.8	37.8	17.5	23.5	26.4	26.9	19.7	28.6
MemoRAG	23.2	33.4	24.5	25.7	25.1	29.8	26.0	29.8	32.6	44.5	27.3	42.0	26.9	33.4	32.7	34.3	27.3	34.1
GraphRAG	22.1	31.6	23.8	29.0	22.2	33.2	24.6	23.9	31.4	44.0	27.2	42.7	24.3	29.2	26.7	26.1	25.3	32.5
Dataset	TECH		NOV		ART		HUM		PAPER		SCI		FIN		LEG		AVE	
LEVEL-4	R-L	$\mathcal{S}$-F1	R-L	$\mathcal{S}$-F1	R-L	$\mathcal{S}$-F1	R-L	$\mathcal{S}$-F1	R-L	$\mathcal{S}$-F1	R-L	$\mathcal{S}$-F1	R-L	$\mathcal{S}$-F1	R-L	$\mathcal{S}$-F1	R-L	$\mathcal{S}$-F1
Top-1	16.3	24.5	13.1	20.4	14.9	9.0	15.9	13.8	17.7	17.4	14.7	15.6	13.0	13.0	11.8	16.5	14.7	16.3
Top-5	16.8	23.4	14.4	26.2	17.7	16.1	15.3	16.6	16.9	15.6	17.7	21.5	11.8	9.9	11.7	16.9	15.3	18.3
Top-10	21.1	40.2	17.4	22.7	20.3	17.1	18.4	22.9	20.5	16.1	18.0	19.3	16.0	19.9	13.8	8.3	18.2	20.8
Top-50	17.8	41.4	16.7	33.9	17.3	32.7	17.5	34.1	17.5	28.0	15.9	41.7	15.9	31.6	16.7	34.1	16.9	34.7
LLM	16.2	35.0	17.4	37.3	16.8	34.4	17.2	29.6	17.1	31.4	15.2	32.2	19.6	35.2	15.6	30.1	16.9	33.2
Lingua-2	13.9	32.2	14.1	23.3	15.0	24.9	13.8	10.5	17.5	27.0	12.9	22.3	20.4	35.8	15.8	25.1	15.4	25.2
MInference	16.2	37.3	18.6	34.5	16.9	37.0	17.7	28.0	17.1	32.4	15.7	28.1	18.8	35.6	15.5	33.6	17.1	33.3
HyDE	16.8	25.6	14.8	20.0	16.7	14.2	15.5	16.9	15.2	16.7	19.4	20.5	13.7	18.6	12.6	15.2	15.6	18.4
RQRAG	16.0	22.1	14.8	17.8	17.6	15.8	16.0	17.0	15.3	17.9	17.7	24.0	13.2	13.1	12.8	11.1	15.4	17.4
MemoRAG	17.7	43.8	20.0	44.1	19.8	37.2	19.7	37.8	20.4	26.1	16.9	36.3	19.8	30.1	17.9	24.2	19.0	35.0
GraphRAG	20.7	37.3	21.1	37.5	22.7	34.4	22.4	31.4	23.7	21.5	20.4	27.4	19.2	20.2	15.1	19.5	20.6	28.7