How Much Reasoning Do Retrieval-Augmented Models Add beyond LLMs? A Benchmarking Framework for Multi-Hop Inference over Hybrid Knowledge

Junhong Lin
junhong@mit.edu
Massachusetts Institute of Technology
Cambridge, MA, United States

Bing Zhang
Bing.Zhang@ibm.com
IBM Research
San Jose, CA, United States

Song Wang
song.wang@ucf.edu
University of Central Florida
Orlando, FL, United States

Ziyan Liu
zl488@cornell.edu
Jersey City, NJ, United States

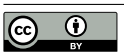
Dan Gutfreund
dgutfre@us.ibm.com
IBM Research
United States

Julian Shun
jshun@mit.edu
Massachusetts Institute of Technology
Cambridge, MA, United States

Yada Zhu
yzhu@us.ibm.com
IBM Research
Yorktown Heights, NY, United States

Abstract

Large language models (LLMs) continue to struggle with knowledge-intensive questions that require up-to-date information and multi-hop reasoning. Augmenting LLMs with hybrid external knowledge, such as unstructured text and structured knowledge graphs, offers a promising alternative to costly continual pretraining. As such, reliable evaluation of their retrieval and reasoning capabilities becomes critical. However, many existing benchmarks increasingly overlap with LLM pretraining data, which means that answers or supporting knowledge may already be encoded in model parameters, making it difficult to distinguish genuine retrieval and reasoning from parametric recall. We introduce HybridRAG-Bench, a contamination-aware benchmark construction framework to evaluate retrieval-intensive, multi-hop reasoning over hybrid knowledge. HybridRAG-Bench automatically couples unstructured text and structured knowledge graph representations derived from recent scientific literature on arXiv, and generates knowledge-intensive question-answer pairs grounded in explicit reasoning paths. The framework supports flexible domain and time-range selection, enabling contamination-aware, customizable evaluation as models and knowledge evolve. Experiments across three domains (artificial intelligence, governance and policy, and bioinformatics) demonstrate that HybridRAG-Bench rewards genuine retrieval and reasoning rather than parametric recall, offering a principled testbed for evaluating hybrid knowledge–augmented reasoning systems. We release our code and data at github.com/junhongmit/HybridRAG-Bench.



KDD '26, Jeju Island, Republic of Korea

ACM ISBN 979-8-4007-2259-2/2026/08
https://doi.org/10.1145/3770855.3817481

CCS Concepts

• **Computing methodologies** → **Natural language processing**; **Knowledge representation and reasoning**; **Machine learning**; • **Information systems** → **Information retrieval**.

Keywords

Large Language Model, Knowledge Graph, Retrieval, Reasoning

ACM Reference Format:
Junhong Lin, Bing Zhang, Song Wang, Ziyan Liu, Dan Gutfreund, Julian Shun, and Yada Zhu. 2026. How Much Reasoning Do Retrieval-Augmented Models Add beyond LLMs? A Benchmarking Framework for Multi-Hop Inference over Hybrid Knowledge. In *Proceedings of the 32nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining V.2 (KDD '26), August 09–13, 2026, Jeju Island, Republic of Korea.* ACM, New York, NY, USA, 12 pages. https://doi.org/10.1145/3770855.3817481

1 Introduction

Large language models (LLMs) increasingly rely on external knowledge to solve knowledge-intensive tasks that require multi-hop reasoning with dispersed evidence. Rather than continually retraining models to absorb newly emerging knowledge, recent systems augment LLMs with external text corpora and structured knowledge graphs (KGs) through retrieval-augmented generation (RAG) and structured knowledge integration (KG-RAG). While these approaches have demonstrated strong empirical gains, a fundamental question remains unresolved: *how much additional **reasoning** do retrieval-augmented models actually contribute beyond the base LLM?*

Recent studies have indicated that *pretraining contamination*, the overlap between benchmark data and model pretraining corpora, can inflate performance and undermine fair comparison across model generations [12, 27, 29, 36, 38, 45]. This issue is severe for hybrid KG-RAG methods, such as Chain-of-Knowledge (CoK) [26], Reasoning-on-Graph (RoG) [32], Think-on-Graph (ToG) [39], and Plan-on-Graph (PoG) [8], which are explicitly designed to incorporate structured retrieval, graph traversal, and multi-hop reasoning.

Table 1: Average accuracy (%) and standard deviation of directly prompting several LLMs with different knowledge cutoff times on the `Movie` and `Sports` Q&A datasets (developed in March 2024), calculated over 5 runs. The best results are highlighted in bold.

LLM (Cutoff Time)	Qwen2-72B (June 2023)	Qwen2.5-72B (October 2023)	Qwen3-32B (mid-2024)
`LMArena` [10]	1263	1303 (+3.1%)	**1347** (+6.7%)
`OpenLLMLeaderboard` [15]	43.59%	47.98% (+4.4%)	–
`Movie` (All)	24.42 ± 0.38	34.27 ± 0.32 (+40.3%)	**43.86 ± 1.00** (+79.6%)
`Sports` (All)	16.17 ± 0.45	23.78 ± 0.20 (+47.1%)	**32.71 ± 0.96** (+102%)
`Movie` (Fast-changing Facts)	5.38 ± 2.11	15.58 ± 0.45 (+189%)	**22.31 ± 1.05** (+315%)
`Sports` (Fast-changing Facts)	3.79 ± 0.34	11.56 ± 0.26 (+205%)	**27.39 ± 0.40** (+622%)

To evaluate these methods meaningfully, benchmarks must require models to retrieve relevant evidence from large external sources and compose multiple facts through reasoning; however, the validity of existing benchmarks is increasingly questionable due to both *pretraining contamination* and the difficulty of *continually constructing well-grounded multi-hop questions at scale* to shed light on models' strengths and weaknesses. Many widely-used multi-hop QA benchmarks, such as HotpotQA (2018) [48], MetaQA (2018) [51], CWQ (2018) [40], TriviaQA (2017) [23], WebQuestions (2013) [4], were constructed before the widespread deployment of LLMs and rely on manual curation over static knowledge sources that widely exist in LLM pretraining corpora, making it difficult to isolate the contribution of retrieval and reasoning modules.

The impact of this overlap is apparent. For example, when asked "*What is the latest film that Denis Villeneuve has been involved in?*", both Qwen2-72B and Qwen2.5-72B (trained in 2023) incorrectly answer *Dune (2021)*, while Qwen3-32B (trained in 2024) correctly answers *Dune: Part Two (2024)*, despite identical prompting and comparable (even smaller) model parameter size. More broadly, we consider a controlled motivating example by **directly prompting** them on two CRAG datasets (`Movie` and `Sports`) [47] developed in March 2024. As shown in Table 1, the QA accuracy increases sharply (up to +102%) with later pretraining cutoffs, while improvements on general reasoning benchmarks remain modest (+6.7%), which cannot solely explain these QA accuracy improvements. When restricting evaluation to questions tagged with involving **fast-changing facts**, earlier models achieve near-zero accuracy. These results indicate that benchmark performance can be dominated by pretraining exposure rather than retrieval or reasoning ability, obscuring genuine methodological progress.

To address this gap, we introduce HybridRAG-Bench, a fully automated benchmark construction framework designed to evaluate retrieval-intensive and multi-hop reasoning under controlled knowledge settings. Rather than proposing a static benchmark tied to a fixed corpus, HybridRAG-Bench is designed as a reusable evaluation framework that can continuously generate new hybrid reasoning benchmarks as domains, corpora, and LLM knowledge boundaries evolve. HybridRAG-Bench explicitly externalizes all knowledge required for question answering, thereby satisfying four key properties for valid evaluation. (1) **Contamination resistance:** HybridRAG-Bench collects scientific corpora from user-specified topics and time ranges on arXiv, reducing overlap with LLM pretraining data and ensuring that correctness depends on retrieval rather than memorization. (2) **Explicit externalized knowledge:** HybridRAG-Bench constructs a hybrid knowledge environment consisting of aligned text chunks and extracted knowledge graphs, making the location, structure, and provenance of required knowledge explicit. (3) **Diverse multi-hop reasoning requirements:** HybridRAG-Bench generates challenging question-answer pairs grounded in explicit reasoning paths that cover single-hop lookup, conditional reasoning, multi-hop inference, hard multi-hop chains, counterfactual reasoning, and open-ended synthesis. (4) **Scalable regeneration across domains and time ranges:** HybridRAG-Bench supports automatic quality control and rapid regeneration of benchmarks across evolving topics, time periods, and knowledge scopes without manual curation. By meeting these properties, HybridRAG-Bench provides a robust and evolving testbed for evaluating retrieval-augmented and multi-hop reasoning systems.

HybridRAG-Bench makes the following contributions:

- **A benchmark construction framework for retrieval-intensive reasoning.** HybridRAG-Bench is a reusable benchmark construction framework for evaluating multi-hop reasoning over hybrid structured and unstructured knowledge.[1] The framework generates diverse, knowledge-intensive questions grounded in latent knowledge graph paths, yielding challenges that require genuine retrieval and reasoning.
- **Diagnostic benchmark instances for RAG and KG-RAG methods.** Using the proposed framework, HybridRAG-Bench instantiates evaluation benchmarks that expose clear and consistent performance gaps across retrieval and reasoning strategies, distinguishing naïve augmentation from effective hybrid reasoning over text and graphs.
- **Controlled evaluation of retrieval versus memorization.** HybridRAG-Bench enables systematic control over domains and document time ranges, allowing principled evaluation of whether performance gains stem from genuine retrieval and multi-hop reasoning rather than parametric memorization.

2 Related Work

KGQA Benchmarks. The Knowledge-Graph Question Answering (KGQA) benchmarks evaluate systems' reasoning and accuracy in handling natural language queries over knowledge graphs. Early benchmarks primarily relied on Freebase [5] and its subsets. WebQuestions [4] provided foundational question-answer pairs, which were re-annotated in WebQSP [49] with semantic parses (SPARQL queries) to allow limiting reasoning to semantically valid paths. CWQ [40] tested reasoning capabilities on questions of increased compositional complexity. GrailQA [18] and KQA Pro [7] further expanded question diversity. Because Freebase was discontinued, much of its data has been ingested by modern LLMs during pretraining. Other datasets like MetaQA [51], LC-QuAD [41], and LC-QuAD 2.0 [13] used collaboratively maintained knowledge graphs such as DBpedia [24] and Wikidata [42]. Recent KG-RAG datasets retrieve and expose graph evidence to LLMs and evaluate the generated natural-language responses. CRAG [47] extends beyond Wikipedia to include public data categorized by query dynamism.

[1]The code and data are available at github.com/junhongmit/HybridRAG-Bench.

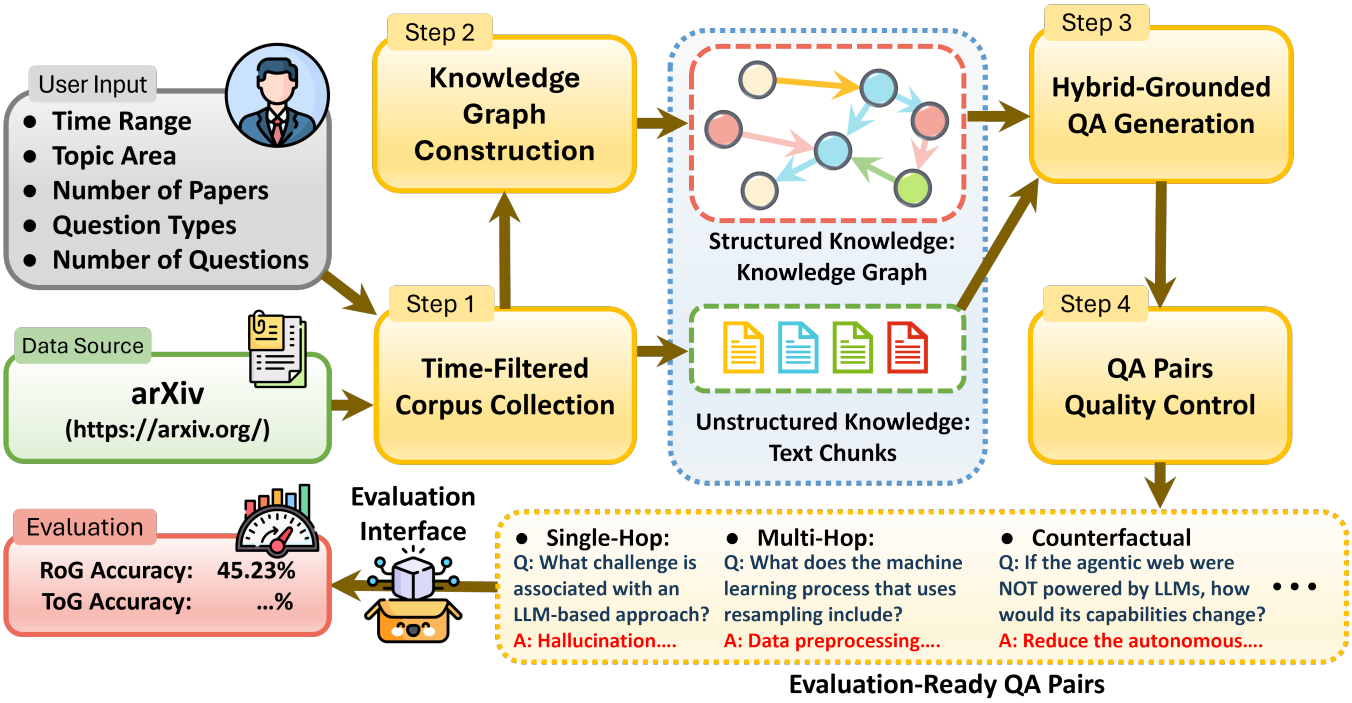


Figure 1: Illustration of the HybridRAG-Bench benchmarking framework.

KGQAGen [50] generates multi-hop QA pairs from local subgraphs and ensures accuracy via SPARQL execution.

As LLMs increasingly incorporate web-scale and scientific corpora during pretraining, KGQA datasets face a growing risk of knowledge overlap with the models' internal memory. Several works have tried to mitigate this issue. For example, Dynamic-KGQA [11] attempts to generate adaptive QA pairs using varied reasoning paths within localized sub-knowledge graphs but it does not provide a definitive isolation from parametric memory in LLMs. Recent work, such as ArxivRoll [29] and OKBench [28], offers dynamic frameworks for generating private evaluation questions from recent documents or news reports; however, these frameworks are not designed for KGQA, as it operates on single-document fragments and does not support graph-structured retrieval or multi-fact reasoning.

Retrieval-Augmented and Knowledge-Grounded LLMs. To overcome the limitations of parametric knowledge and avoid the cost of continual pretraining, a large body of work has explored augmenting LLMs with external knowledge through retrieval-augmented generation (RAG) or from knowledge graphs. Text-based RAG [2, 25] methods retrieve unstructured documents to improve factual accuracy, while KG-based approaches leverage structured graphs for relational and multi-hop reasoning. Specifically, recent KG-based reasoning methods adopt different retrieval and reasoning paradigms. CoK [26] iteratively retrieves external knowledge from unstructured text, knowledge graphs, and tables into subsequent reasoning steps; RoG [32] relies on a question-specific subgraph and allows self-correcting erroneous reasoning paths; ToG [39] and ToG2.0 [33] optimize retrieval by performing path expansion and pruning based on seed entities and context; and PoG [8] and R2-KG [22] improve answer reliability by enhancing LLM-based judging during exploration and answering. To reason over temporal facts (e.g., TimeQuestions [21] and MultiTQ [9]), EvoReasoner [30] includes relation and entity reranking based on their temporal alignment with reasoning subgoals.

While these frameworks offer diverse advantages in relational reasoning and hallucination reduction, they are primarily evaluated using static benchmarks such as WebQSP [49], CWQ [40], GrailQA [18], HotpotQA [48], and MetaQA [51]. These benchmarks are constructed once from Freebase [5], DBpedia [24], and Wikidata [42]. Consequently, their reported performance can be overestimated due to the overlap between the models' internal parametric knowledge and the external knowledge used for evaluation.

3 Problem Definition and Preliminaries

We formalize the problem addressed by HybridRAG-Bench as *contamination-aware evaluation of knowledge-intensive question answering over hybrid knowledge composed of unstructured text and structured knowledge graphs.* Evaluation is performed within a single domain at a time, while the framework supports instantiation across different domains and time ranges.

3.1 Domains, Corpora, and Knowledge Graphs

Let $\{\mathcal{D}^{(1)}, \ldots, \mathcal{D}^{(K)}\}$ denote document corpora corresponding to K distinct domains. Each domain m is specified by a set of document

selection criteria (e.g., subject categories and keyword constraints), which define the scope of domain-relevant documents.

For each domain m, documents are collected within a user-specified time window $[T^{(m)}_{\text{start}}, T^{(m)}_{\text{end}}]$, forming the corpus and its induced constructed knowledge graph:

$$\mathcal{D}^{(m)} = \{d^{(m)}_1, \ldots, d^{(m)}_{N_m}\}, \quad \mathcal{G}^{(m)} = (V^{(m)}, E^{(m)}, \pi^{(m)}),$$

The time window serves as a mechanism for controlling the temporal scope of the knowledge source and reducing overlap with LLM pretraining data. Each knowledge graph is constructed independently per domain and may differ in entity types, relation schemas, graph topology, and temporal dynamics. No entities or relations are shared across domains.

3.2 Questions and Evaluation Scope

For each domain m, we define a set of question-answer pairs

$$Q^{(m)} = \{(q_i, t_i, a_i)\},$$

where q_i is a question issued at time t_i and a_i is the corresponding ground-truth answer. Each question is evaluated exclusively with respect to the domain-specific knowledge graph snapshot $\mathcal{G}^{(m)}_{t_i}$ and documents available up to time t_i.

We assume that all documents, knowledge graph facts, and answers are strictly newer than the pretraining cutoff of the evaluated large language models. Consequently, correct answers cannot be recovered via parametric memorization alone.

3.3 Task Definition

Given a question $q \in Q^{(m)}$ issued at time t_q, the task is to predict the answer a using information available within the domain-specific hybrid knowledge environment. A model f may reason over: (1) the knowledge graph snapshot $\mathcal{G}^{(m)}$, and (2) a set of retrieved documents $\mathcal{D}^{(m)}_q \subseteq \mathcal{D}^{(m)}$. Formally, the model produces

$$\hat{a} = f\left(q, \mathcal{G}^{(m)}, \mathcal{D}^{(m)}_q\right),$$

which is evaluated against the ground-truth answer a.

4 Benchmark Construction

As illustrated in Figure 1, HybridRAG-Bench is a fully automated framework for constructing benchmarks that evaluate retrieval-intensive, multi-hop reasoning over hybrid knowledge. Concretely, HybridRAG-Bench follows a four-stage pipeline. First, given user-specified topic filters and time ranges, the framework collects time-filtered arXiv corpora that define the external knowledge source available to retrieval-based methods. Second, it constructs hybrid knowledge environments by extracting aligned unstructured text chunks and structured knowledge graphs from the corpus, enabling both document retrieval and graph-based traversal. Third, the framework generates diverse question-answer pairs grounded in explicit reasoning paths and supporting evidence, covering single-hop, conditional, multi-hop, hard multi-hop, counterfactual, and open-ended reasoning. Finally, an automated quality control stage filters and validates generated questions to ensure answerability, independence from document-specific phrasing, and non-redundancy.

Together, these components enable reproducible construction of retrieval-intensive benchmarks for evaluating RAG and KG-RAG methods under controlled knowledge settings. We next describe each stage in detail.

4.1 Time-Filtered Corpus Collection

For each domain, we collect a corpus of arXiv papers based on user-specified document selection criteria, including subject categories (e.g., `cs.AI` and `cs.LG`) and optional keyword constraints (e.g., *reinforcement learning*). This mechanism allows HybridRAG-Bench to be instantiated flexibly across domains while maintaining topical coherence.

Document collection is performed over a configurable time window, enabling evaluation under different knowledge cutoffs. In our experiments, we select document windows that postdate the public pretraining cutoffs of contemporary LLMs, reducing the likelihood that answers can be recovered via parametric memorization alone. For each paper, the parsed text, section-level segmentation (e.g., abstract, introduction, and methods), and metadata, including title, authors, categories, and submission timestamps, are stored. These documents serve as the unstructured knowledge source for retrieval-based methods.

4.2 Knowledge Graph Construction

From the collected corpus, we construct a knowledge graph using EvoKG [30], a document-driven framework for extracting and organizing structured knowledge from unstructured text. The resulting graph captures entities (e.g., methods, datasets, tasks, and policies) and relations expressed in the literature, and is constructed independently for each domain.

Entity Extraction and Alignment. Given a batch of documents, EvoKG applies a large language model to extract candidate entities and relations. Extracted entities may exhibit lexical variation (e.g., abbreviated method names), semantic ambiguity, or partial descriptions. To address this, EvoKG performs context-aware entity alignment, matching each extracted entity against existing KG nodes using joint embeddings over entity type, name, and description. If no existing node exceeds a similarity threshold, a new entity is created; otherwise, the extracted mention is merged with the matched node, preserving provenance information. This alignment step prevents entity fragmentation and ensures consistency across documents and time.

Relation Normalization and Evidence Tracking. Extracted relations are normalized to a domain-specific schema and linked to supporting textual evidence. Rather than enforcing a single canonical fact, the knowledge graph retains multiple candidate relations when supported by the corpus, along with confidence scores derived from frequency, recency, and textual support. This design allows the graph to represent uncertainty and variation commonly observed in scientific writing.

The resulting knowledge graph provides structured relational scaffolding that complements the unstructured document corpus, enabling hybrid retrieval and reasoning.

Table 2: Question statistics by types.

Question Type	Arxiv-AI	Arxiv-CY	Arxiv-BIO
Single-hop	249 (29%)	238 (25%)	264 (25%)
Single-hop w/ conditions	139 (16%)	128 (13%)	193 (19%)
Multi-hop (regular)	165 (19%)	149 (15%)	195 (19%)
Multi-hop (difficult)	149 (17%)	166 (17%)	139 (13%)
Counterfactual	114 (13%)	173 (18%)	59 (6%)
Open-ended	47 (5%)	112 (12%)	190 (18%)
All	863	966	1040

4.3 Hybrid-Grounded Question-Answer Generation

HybridRAG-Bench constructs benchmark questions by grounding them in both *structured reasoning paths from the knowledge graph* and *supporting unstructured textual evidence from the document corpus*. This hybrid grounding ensures that questions cannot be answered solely by graph traversal or document retrieval alone, and instead require effective integration of both modalities.

Reasoning Path Sampling. We first sample reasoning paths from the knowledge graph of the form

$$p = (v_0 \xrightarrow{r_1} v_1 \xrightarrow{r_2} \dots \xrightarrow{r_k} v_k),$$

where v_k is the answer entity. Each sampled path defines a valid relational structure connecting the question subject to its answer. For every entity and relation along the path, we retrieve the associated supporting text chunks from the corpus, which provide contextual descriptions, qualifiers, and evidence.

Hybrid Question Construction. For all question types, we condition a large language model on a structured context that includes: (1) a sampled reasoning path from the knowledge graph, which specifies the relational constraints, (2) textual evidence associated with the entities and relations along the path, and (3) a small set of in-context examples illustrating the desired question format. The reasoning path serves as a structural scaffold, while the textual context guides natural language formulation and enables questions that require interpretation or synthesis beyond symbolic lookup. The model is instructed to generate a question that is faithful to the provided evidence and to produce a ground-truth answer that can be verified from the same context. If the available evidence is insufficient to support a well-posed question (e.g., missing attributes or ambiguous relations), the model is explicitly instructed to abstain from generating an example.

Using this hybrid grounding mechanism, we generate a diverse set of question-answer pairs. The distribution of question types across domains is summarized in Table 2.

- **Single-hop questions.** These questions correspond to a single relation lookup (v_0, r, v_1). The generated question directly queries an attribute or relation of the subject entity, while the answer is the terminal entity or value supported by both the KG edge and its textual evidence.
- **Single-hop questions with conditions.** In addition to a single KG relation, these questions include explicit constraints extracted from textual descriptions, such as temporal qualifiers (e.g., "in 2018"), categorical restrictions, or contextual conditions. The conditions are derived from attributes or modifiers mentioned in the supporting text, requiring models to align symbolic relations with unstructured qualifiers.
- **Multi-hop questions**. These questions are grounded in reasoning paths with $k \geq 2$ relations. Answering them requires composing multiple relational steps and retrieving evidence across intermediate entities. The questions are formulated to obscure the intermediate nodes, requiring implicit multi-hop inference.
- **Difficult multi-hop questions**. To increase difficulty, we preferentially sample paths that traverse high-degree entities in the knowledge graph. Such entities introduce large candidate spaces and increase the likelihood of spurious retrieval, placing greater demands on both retrieval precision and reasoning robustness.
- **Counterfactual questions**. Counterfactual questions are constructed by minimally perturbing a relation or attribute in the original reasoning path (e.g., replacing a method, dataset, or condition) and asking how the outcome or conclusion would change. The generated answers are constrained to provide cautious, evidence-based reasoning without introducing unsupported facts.
- **Open-ended questions**. Open-ended questions require synthesizing explanations or summaries from multiple pieces of textual evidence aligned with a reasoning path. Rather than returning a single entity, the answer is a short, evidence-grounded natural language response that integrates information from several nodes and relations.

4.4 QA Pairs Quality Control

Lastly, we apply a multi-stage quality control process to ensure that all generated question-answer pairs in HybridRAG-Bench are well-posed, answerable, and non-redundant.

Answerability and Faithfulness. Using an LLM-as-a-Judge protocol [17], we verify that each question can be answered solely from the provided hybrid context (KG paths and supporting text), and that the ground-truth answer is faithful to that evidence. Questions requiring external knowledge or unsupported inference are removed.

Context Independence and Clarity. We filter out questions that contain document-local references (e.g., "in this paper", "Theorem 12") or are ambiguous or poorly phrased. Retained questions are lightly normalized (e.g., lowercasing and punctuation standardization) without altering their semantic content.

Only question-answer pairs that pass all checks are included in the final benchmark.

4.5 Benchmark Validation

We further conduct human evaluation to assess the quality of the constructed QA pairs. We randomly sample 100 stratified questions per domain covering all question types, and ask three independent annotators from diverse backgrounds (industry researcher, undergraduate student, and graduate student) to evaluate each sample in terms of answerability, answer correctness, and question clarity.

Table 3: Human evaluation of benchmark quality. Results are averaged across three domains using 100 stratified QA samples per domain.

Metric	Results	Agreement
Answerability	96%	$89.7_{\pm 1.7}$
Clarity	94%	$88.7_{\pm 0.5}$
Correctness	92%	$80.3_{\pm 1.2}$

As shown in Table 3, 96% of questions are judged answerable, and 94% are clearly phrased, while the overall error rate is only 8% (i.e., correctness rate is 92%). Pairwise agreement across annotators is consistently high, indicating stable annotation quality. Manual inspection shows that most errors arise from ambiguous or multi-answer questions rather than incorrect grounding.

Importantly, we also validate the reasoning requirements of the benchmark. All sampled multi-hop questions are confirmed to require genuine multi-step reasoning rather than reducible single-hop retrieval, and we observe diverse reasoning patterns instead of templated path structures.

5 Experiment

We design our experiments to evaluate HybridRAG-Bench as a retrieval-intensive benchmark for knowledge-grounded reasoning and to assess whether it meaningfully differentiates between LLM-only, RAG-based, and KG-RAG reasoning approaches. In particular, we seek to answer the following research questions.

- **RQ1:** Are HybridRAG-Bench questions genuinely challenging across LLM scales?
- **RQ2:** Does HybridRAG-Bench require external retrieval to answer questions correctly?
- **RQ3:** Does structured knowledge (KG) provide complementary value beyond text-only retrieval?
- **RQ4:** Does HybridRAG-Bench discriminate between different retrieval and reasoning strategies?

5.1 Experimental Setup

Benchmark Instantiation. We instantiate HybridRAG-Bench on three domain-specific corpora collected from recent arXiv papers: AI (reinforcement learning), CY (governance and policy), and BIO (bioinformatics). For each domain, we construct a domain-specific knowledge graph and generate approximately 1,000 question-answer pairs grounded in explicit knowledge graph paths and supporting textual evidence. As shown in Table 2, the questions cover single-hop, conditional single-hop, multi-hop, hard multi-hop, counterfactual, and open-ended reasoning. We manually inspect a subset of the generated data to verify question quality and faithfulness. For answer evaluation, we adopt an LLM-as-a-Judge protocol following CRAG [47], using the same judge prompt and evaluation models to assess answer correctness and faithfulness.

Baselines. We use four state-of-the-art LLMs with different parameter sizes and knowledge cutoff, including DeepSeek V3.2 (685B, Jul 2024) [31], LLaMA 4-Maverick (400B, Aug 2024) [1], Qwen 3-VL (235B, mid-to-late 2025) [3], Qwen 2.5 (72B, Oct 2023) [46], LLaMA 3.3 (70B, December 2023), LLaMA 3.1 (8B, Dec 2023) [16], and GPT-5 (Nano, Sep 2024) [37]. We compare a broad range of baselines that differ in how they retrieve and integrate external knowledge. **(1) LLM-only prompting**: these methods do not access external knowledge, and include IO-prompt (IO) [6], Chain-of-Thought (CoT) [44], and Self-Consistency (SC) [43]. **(2) Text-based RAG**: these methods retrieve relevant passages from the full arXiv corpus using dense retrieval, and include Retrieval-Augmented Generation (RAG) [25]. **(3) Graph-indexed RAG**: these methods augment text retrieval with an internal graph-based indexing structure over the corpus, while still operating solely on unstructured text rather than an explicit external knowledge graph. This category includes HippoRAG2.0 [20], which organizes retrieved evidence through a latent memory graph, GraphRAG [14], which builds entity-relation graphs from text chunks for retrieval expansion, and LightRAG [19], which constructs lightweight graph-based retrieval indices for efficient multi-hop evidence collection. **(4) Naive KG augmentation**: these methods directly inject local graph neighborhoods (e.g., one-hop neighbors) into the prompt, and include 1-hop KG [47] (augmenting the LLM with facts from 1-hop KG neighbors of the topic entities) and RAG + 1-hop KG [47]. **(5) Hybrid KG-RAG**: these methods jointly leverage graph structure and textual evidence through structured retrieval and multi-hop reasoning, and include Chain-of-Knowledge (CoK) [26], Reasoning-on-Graph (RoG) [32], Think-on-Graph (ToG) [39], Think-on-Graph 2.0 (ToG2.0) [33], Plan-on-Graph (PoG) [8], R2-KG [22], and EvoReasoner [30]. We report the average and standard deviation over 5 runs for all models.

5.2 Experiment Results

RQ1: Are HybridRAG-Bench Questions Challenging Across LLM Scales? As shown in Table 5, across all three domains, LLM-only prompting achieves consistently low accuracy across all three domains, ranging from 23–40%, even when scaling model size from LLaMA-3.1-8B to DeepSeek-V3.2-685B. Larger models yield modest improvements, but performance remains low. This trend indicates that HybridRAG-Bench questions cannot be reliably answered through parametric knowledge alone, even by frontier-scale LLMs. The limited gains from model scaling suggest that success on HybridRAG-Bench does not primarily stem from memorization or general language competence, but instead requires access to external knowledge and non-trivial reasoning. Overall, these results confirm that HybridRAG-Bench poses a substantial and persistent challenge across LLM scales.

RQ2: Does HybridRAG-Bench Require External Retrieval? Introducing external retrieval leads to large and consistent performance gains. As observed in Table 5, for example, text-based RAG improves accuracy by 7–29% over LLM-only prompting across most model–domain combinations, with the exception of DeepSeek-V3.2 on `Arxiv-CY`. These gains demonstrate that external knowledge is critical for answering HybridRAG-Bench questions. Without retrieval, models fail to obtain the domain-specific and up-to-date information required by the benchmark. Notably, naïve KG-based augmentation can *degrade* performance relative to LLM-only prompting. Directly injecting one-hop graph neighborhoods without selective retrieval or structured reasoning introduces substantial noise,

Table 4: Experiment results (accuracy, %) across LLMs of different scales. We highlight the first and second best results.

Datasets →	Arxiv-AI							Arxiv-CY						
Methods↓	DeepSeek3.2 685B	LLaMA4 Maverick 400B	Qwen3-VL 235B	Qwen2.5 72B	LLaMA3.3 70B	LLaMA3.1 8B	GPT-5 Nano	DeepSeek3.2 685B	LLaMA4 Maverick 400B	Qwen3-VL 235B	Qwen2.5 72B	LLaMA3.3 70B	LLaMA3.1 8B	GPT-5 Nano
IO	$37.75_{\pm 0.64}$	$39.01_{\pm 0.57}$	$30.36_{\pm 0.31}$	$27.07_{\pm 0.47}$	$34.11_{\pm 0.67}$	$23.08_{\pm 0.64}$	$19.66_{\pm 1.24}$	$43.62_{\pm 0.62}$	$42.41_{\pm 0.22}$	$30.78_{\pm 0.62}$	$27.83_{\pm 0.20}$	$35.32_{\pm 0.17}$	$24.06_{\pm 0.28}$	$19.91_{\pm 1.21}$
CoT	$36.69_{\pm 0.31}$	$33.41_{\pm 0.44}$	$36.54_{\pm 0.18}$	$35.32_{\pm 0.35}$	$35.32_{\pm 0.58}$	$27.65_{\pm 0.63}$	$33.49_{\pm 0.35}$	$39.50_{\pm 0.75}$	$34.61_{\pm 0.42}$	$40.20_{\pm 0.59}$	$39.01_{\pm 0.63}$	$40.64_{\pm 0.59}$	$28.47_{\pm 1.04}$	$32.16_{\pm 1.05}$
SC	$35.41_{\pm 0.63}$	$32.52_{\pm 0.24}$	$33.37_{\pm 4.93}$	$34.02_{\pm 0.72}$	$36.27_{\pm 0.88}$	$26.95_{\pm 0.85}$	$30.82_{\pm 1.29}$	$36.89_{\pm 1.49}$	$34.27_{\pm 0.45}$	$38.75_{\pm 0.33}$	$37.62_{\pm 0.63}$	$41.24_{\pm 0.42}$	$29.75_{\pm 0.70}$	$30.37_{\pm 0.12}$
RAG	$43.68_{\pm 0.93}$	$47.16_{\pm 0.16}$	$49.71_{\pm 0.24}$	$50.44_{\pm 0.07}$	$47.30_{\pm 0.15}$	$38.32_{\pm 0.27}$	$41.83_{\pm 0.20}$	$41.82_{\pm 0.21}$	$45.50_{\pm 0.22}$	$45.34_{\pm 0.15}$	$40.99_{\pm 4.94}$	$42.99_{\pm 0.23}$	$35.56_{\pm 0.07}$	$45.65_{\pm 0.15}$
1-hop KG	$27.55_{\pm 0.31}$	$35.46_{\pm 0.35}$	$24.87_{\pm 0.13}$	$22.69_{\pm 0.19}$	$26.19_{\pm 0.34}$	$22.57_{\pm 0.38}$	$22.98_{\pm 0.99}$	$29.63_{\pm 0.28}$	$34.82_{\pm 0.26}$	$25.53_{\pm 0.06}$	$21.86_{\pm 0.24}$	$23.48_{\pm 0.20}$	$23.56_{\pm 0.70}$	$21.05_{\pm 0.61}$
RAG+1-hop KG	$49.42_{\pm 0.08}$	$54.52_{\pm 0.08}$	$52.14_{\pm 0.16}$	$50.64_{\pm 0.20}$	$51.49_{\pm 0.18}$	$43.68_{\pm 0.12}$	$46.23_{\pm 0.16}$	$47.27_{\pm 0.52}$	$52.95_{\pm 0.22}$	$50.05_{\pm 0.66}$	$47.48_{\pm 0.16}$	$45.69_{\pm 1.66}$	$42.55_{\pm 0.29}$	$40.37_{\pm 0.40}$
CoK	$23.08_{\pm 0.59}$	$45.96_{\pm 0.18}$	$46.66_{\pm 0.18}$	$42.69_{\pm 1.33}$	$38.49_{\pm 0.80}$	$35.62_{\pm 2.61}$	$36.69_{\pm 0.84}$	$27.23_{\pm 0.93}$	$49.24_{\pm 0.73}$	$48.45_{\pm 1.36}$	$40.43_{\pm 0.69}$	$39.01_{\pm 0.74}$	$36.89_{\pm 0.97}$	$32.37_{\pm 1.12}$
RoG	$34.79_{\pm 0.38}$	$47.32_{\pm 0.59}$	$46.89_{\pm 0.24}$	$37.91_{\pm 0.62}$	$43.55_{\pm 0.36}$	$25.68_{\pm 0.48}$	$42.64_{\pm 0.58}$	$40.81_{\pm 0.38}$	$49.90_{\pm 1.45}$	$50.28_{\pm 0.49}$	$40.91_{\pm 0.14}$	$44.24_{\pm 0.48}$	$26.15_{\pm 1.25}$	$43.89_{\pm 2.29}$
ToG	$40.69_{\pm 0.78}$	$40.29_{\pm 1.77}$	$42.10_{\pm 0.81}$	$44.50_{\pm 0.40}$	$39.91_{\pm 0.23}$	$17.54_{\pm 1.22}$	$8.34_{\pm 0.09}$	$39.34_{\pm 0.45}$	$38.41_{\pm 0.31}$	$34.99_{\pm 0.27}$	$37.33_{\pm 0.25}$	$34.58_{\pm 0.56}$	$15.65_{\pm 2.64}$	$8.80_{\pm 0.10}$
ToG2.0	$63.59_{\pm 0.80}$	$58.86_{\pm 0.51}$	$59.64_{\pm 1.14}$	$58.61_{\pm 0.31}$	$59.37_{\pm 0.53}$	$49.25_{\pm 0.51}$	$58.05_{\pm 0.55}$	$66.17_{\pm 0.47}$	$60.87_{\pm 0.27}$	$62.84_{\pm 0.82}$	$59.05_{\pm 0.45}$	$60.85_{\pm 0.31}$	$51.91_{\pm 0.84}$	$51.76_{\pm 0.60}$
PoG	$38.70_{\pm 0.27}$	$37.47_{\pm 0.27}$	$40.64_{\pm 0.26}$	$43.13_{\pm 0.38}$	$39.86_{\pm 0.22}$	$16.55_{\pm 1.05}$	$6.37_{\pm 0.08}$	$37.37_{\pm 0.50}$	$35.47_{\pm 0.49}$	$34.85_{\pm 0.59}$	$35.94_{\pm 0.40}$	$34.02_{\pm 0.32}$	$15.78_{\pm 2.04}$	$5.07_{\pm 0.07}$
R2-KG	$49.01_{\pm 0.99}$	$37.20_{\pm 1.71}$	$43.88_{\pm 0.24}$	$42.97_{\pm 0.92}$	$46.51_{\pm 0.42}$	$25.93_{\pm 1.55}$	$11.24_{\pm 0.47}$	$42.67_{\pm 0.91}$	$38.51_{\pm 1.26}$	$41.17_{\pm 0.51}$	$43.81_{\pm 0.70}$	$46.07_{\pm 0.53}$	$27.76_{\pm 1.55}$	$10.25_{\pm 0.36}$
GraphRAG	$70.94_{\pm 0.61}$	$60.79_{\pm 0.62}$	**$74.20_{\pm 0.27}$**	$69.06_{\pm 1.44}$	$63.94_{\pm 1.19}$	$44.40_{\pm 0.61}$	$54.04_{\pm 0.35}$	$60.95_{\pm 0.94}$	$61.12_{\pm 0.59}$	**$66.46_{\pm 0.54}$**	$69.36_{\pm 0.88}$	$64.33_{\pm 1.03}$	$47.87_{\pm 0.85}$	$58.39_{\pm 0.42}$
LightRAG	$70.27_{\pm 0.10}$	$68.60_{\pm 0.08}$	$69.41_{\pm 0.70}$	$59.21_{\pm 0.31}$	$58.82_{\pm 8.06}$	$54.95_{\pm 0.35}$	**$66.74_{\pm 0.40}$**	$61.86_{\pm 0.09}$	**$68.10_{\pm 0.11}$**	$64.35_{\pm 0.33}$	$47.29_{\pm 0.19}$	$62.57_{\pm 1.03}$	$54.20_{\pm 0.34}$	$64.39_{\pm 0.38}$
HippoRAG2.0	$62.39_{\pm 0.13}$	$53.23_{\pm 0.07}$	$57.59_{\pm 0.20}$	$37.71_{\pm 0.06}$	$56.11_{\pm 0.47}$	$47.86_{\pm 0.12}$	$33.41_{\pm 0.07}$	$58.57_{\pm 0.09}$	$50.28_{\pm 0.16}$	$48.21_{\pm 0.12}$	$35.49_{\pm 0.09}$	$56.00_{\pm 0.59}$	$48.90_{\pm 0.21}$	$28.95_{\pm 0.22}$
EvoReasoner	**$75.69_{\pm 0.55}$**	**$74.43_{\pm 0.35}$**	$62.83_{\pm 0.35}$	**$71.30_{\pm 1.36}$**	**$66.78_{\pm 0.67}$**	**$55.74_{\pm 0.82}$**	$64.08_{\pm 1.02}$	**$69.15_{\pm 0.59}$**	$67.22_{\pm 0.83}$	$63.44_{\pm 3.14}$	**$66.89_{\pm 6.13}$**	**$66.31_{\pm 0.75}$**	**$55.22_{\pm 0.28}$**	**$66.73_{\pm 0.87}$**

Table 5: Experiment results (accuracy, %), continued.

Datasets →	Arxiv-BIO						
Methods↓	DeepSeek3.2 685B	LLaMA4 Maverick 400B	Qwen3-VL 235B	Qwen2.5 72B	LLaMA3.3 70B	LLaMA3.1 8B	GPT-5 Nano
IO	$39.88_{\pm 0.56}$	$39.58_{\pm 0.58}$	$30.19_{\pm 0.58}$	$22.81_{\pm 0.16}$	$34.63_{\pm 0.08}$	$25.13_{\pm 0.66}$	$22.82_{\pm 2.44}$
CoT	$39.75_{\pm 0.91}$	$37.56_{\pm 0.47}$	$37.44_{\pm 2.95}$	$36.33_{\pm 0.08}$	$37.46_{\pm 0.55}$	$29.96_{\pm 0.63}$	$32.15_{\pm 1.59}$
SC	$37.29_{\pm 0.90}$	$36.57_{\pm 0.45}$	$33.01_{\pm 8.42}$	$35.15_{\pm 0.64}$	$37.19_{\pm 0.86}$	$29.35_{\pm 0.86}$	$29.01_{\pm 0.53}$
RAG	$48.61_{\pm 0.75}$	$52.40_{\pm 0.41}$	$52.40_{\pm 0.27}$	$52.16_{\pm 0.07}$	$49.18_{\pm 0.48}$	$43.85_{\pm 0.14}$	$50.29_{\pm 0.24}$
1-hop KG	$40.67_{\pm 0.66}$	$40.54_{\pm 0.31}$	$31.76_{\pm 0.15}$	$25.15_{\pm 0.19}$	$29.75_{\pm 0.20}$	$28.77_{\pm 0.28}$	$26.70_{\pm 0.68}$
RAG+1-hop KG	$58.99_{\pm 0.34}$	$59.71_{\pm 0.14}$	$59.42_{\pm 0.27}$	$57.16_{\pm 0.07}$	$54.57_{\pm 0.07}$	$51.20_{\pm 0.61}$	$53.37_{\pm 0.18}$
CoK	$32.04_{\pm 0.89}$	$48.59_{\pm 0.72}$	$49.52_{\pm 0.76}$	$41.21_{\pm 5.22}$	$39.90_{\pm 0.34}$	$35.98_{\pm 0.67}$	$34.78_{\pm 2.89}$
RoG	$36.96_{\pm 0.73}$	$47.53_{\pm 0.24}$	$47.28_{\pm 0.36}$	$38.67_{\pm 0.99}$	$40.02_{\pm 0.16}$	$24.21_{\pm 1.18}$	$36.83_{\pm 1.00}$
ToG	$44.15_{\pm 0.34}$	$39.84_{\pm 0.29}$	$41.99_{\pm 0.06}$	$44.69_{\pm 0.29}$	$42.23_{\pm 0.34}$	$16.06_{\pm 5.88}$	$10.67_{\pm 0.13}$
ToG2.0	$64.69_{\pm 0.95}$	$61.67_{\pm 0.36}$	$61.83_{\pm 0.96}$	$60.36_{\pm 0.39}$	$57.40_{\pm 8.39}$	$51.58_{\pm 1.83}$	$55.58_{\pm 0.43}$
PoG	$42.13_{\pm 0.71}$	$40.70_{\pm 0.45}$	$41.92_{\pm 0.42}$	$44.10_{\pm 0.28}$	$41.79_{\pm 0.34}$	$21.23_{\pm 2.30}$	$6.35_{\pm 0.09}$
R2-KG	$50.15_{\pm 0.73}$	$35.22_{\pm 0.96}$	$46.54_{\pm 0.79}$	$44.25_{\pm 3.21}$	$48.73_{\pm 0.59}$	$27.65_{\pm 1.26}$	$9.13_{\pm 0.39}$
GraphRAG	$68.33_{\pm 0.44}$	$65.94_{\pm 0.60}$	**$74.74_{\pm 0.48}$**	$69.56_{\pm 2.47}$	$68.35_{\pm 0.83}$	$44.02_{\pm 0.52}$	$54.42_{\pm 0.55}$
LightRAG	$69.84_{\pm 0.23}$	$71.90_{\pm 0.13}$	$63.01_{\pm 0.22}$	$56.63_{\pm 0.10}$	$61.98_{\pm 9.32}$	$58.29_{\pm 0.42}$	$66.25_{\pm 0.52}$
HippoRAG2.0	$62.19_{\pm 0.11}$	$55.24_{\pm 0.58}$	$51.79_{\pm 0.11}$	$47.25_{\pm 0.11}$	$56.69_{\pm 0.20}$	$50.33_{\pm 0.09}$	$38.27_{\pm 0.10}$
EvoReasoner	**$75.92_{\pm 0.72}$**	**$75.13_{\pm 0.86}$**	$57.37_{\pm 2.76}$	**$73.31_{\pm 0.55}$**	**$69.94_{\pm 0.36}$**	**$58.13_{\pm 1.07}$**	**$68.37_{\pm 0.41}$**

distracting the model from relevant evidence. This highlights that HybridRAG-Bench does not merely reward access to more information, but specifically requires *effective retrieval* and evidence selection.

RQ3: Does Structured Knowledge Complement Text-Based Retrieval? Methods that jointly leverage structured graph information and unstructured text consistently outperform text-only RAG baselines across all domains (Table 5). While text-based RAG is effective for retrieving descriptive or explanatory information, it struggles with questions that require relational reasoning, entity disambiguation, or multi-hop composition. Structured knowledge graphs provide explicit relational scaffolding that complements unstructured retrieval, enabling more reliable traversal and reasoning. The consistent advantage of hybrid methods over text-only

Table 6: Qwen2.5-72B Model performance (accuracy, %) grouped by question types. We highlight the first and second best results. Due to space limitations, we have omitted the standard deviations, but provide them in Appendix B.

Datasets →	Arxiv-AI						Arxiv-CY						Arxiv-BIO					
Methods↓	*Single-Hop*	*Single-Hop w/ Conditions*	*Multi-Hop*	*Multi-Hop (Difficult)*	*Counterfactual*	*Open-Ended*	*Single-Hop*	*Single-Hop w/ Conditions*	*Multi-Hop*	*Multi-Hop (Difficult)*	*Counterfactual*	*Open-Ended*	*Single-Hop*	*Single-Hop w/ Conditions*	*Multi-Hop*	*Multi-Hop (Difficult)*	*Counterfactual*	*Open-Ended*
IO	21.20	20.29	16.36	17.72	68.07	45.96	17.96	16.02	15.10	15.36	68.64	34.15	19.70	19.38	7.35	16.81	56.27	40.53
CoT	25.22	29.35	25.94	23.76	87.02	50.64	27.48	27.34	24.83	20.96	91.91	40.71	32.20	35.34	18.06	22.61	83.39	57.26
SC	25.46	26.76	24.00	22.68	84.04	50.64	26.30	27.19	23.09	21.45	87.05	40.54	29.55	33.47	19.29	21.01	83.05	56.42
RAG	53.55	67.63	41.82	29.31	48.54	85.11	51.68	68.36	35.23	23.49	23.41	72.32	51.70	68.91	34.69	22.10	40.68	79.21
1-hop KG	33.98	23.88	17.58	14.90	8.07	37.45	43.03	24.22	12.62	10.36	1.73	34.64	35.61	20.52	17.35	17.54	0.00	36.74
RAG + 1-hop KG	58.77	65.47	42.22	32.21	37.72	82.98	65.97	68.75	38.03	28.51	15.03	74.70	61.36	69.17	35.20	37.68	35.59	82.63
CoK	37.19	41.01	33.21	31.81	74.74	66.81	34.45	35.00	28.19	23.61	69.83	55.18	42.58	43.63	25.51	29.71	71.53	62.00
RoG	37.43	32.66	27.64	30.87	71.23	33.62	47.98	32.81	25.77	18.19	71.45	41.79	38.79	38.03	22.86	29.13	77.97	50.21
ToG	53.09	35.11	38.79	39.60	45.09	60.85	57.14	40.31	29.53	25.78	13.99	55.00	45.83	40.28	32.78	45.29	45.34	58.68
ToG2.0	59.52	54.82	44.73	43.89	90.35	83.40	63.61	55.78	29.53	29.16	96.07	69.46	60.08	58.24	38.98	51.88	93.56	80.84
PoG	51.65	36.12	30.55	35.70	53.68	60.85	54.62	38.91	28.19	19.40	20.92	50.89	46.21	38.13	29.80	42.17	60.00	58.42
R2KG	43.13	37.99	36.36	30.34	73.68	45.53	50.34	42.50	30.87	26.14	60.92	48.39	50.53	39.38	35.31	32.32	64.07	53.37
HippoRAG2.0	17.67	29.78	20.48	5.23	23.68	30.21	13.45	19.53	8.59	4.58	14.34	14.29	30.30	41.45	22.45	11.88	56.27	46.11
EvoReasoner	71.75	75.54	65.86	53.47	88.30	90.78	69.75	63.28	54.50	49.04	83.12	82.86	75.15	77.82	56.94	56.09	91.53	89.89

RAG indicates that explicit graph structure helps models organize, connect, and reason over retrieved evidence more effectively. These results confirm that HybridRAG-Bench is sensitive to how models integrate text and graph evidence, making it a suitable testbed for evaluating structured reasoning capabilities in KG-RAG systems.

RQ4: Does HybridRAG-Bench Discriminate Between Retrieval and Reasoning Strategies? For clarity of analysis, we present per-question-type results (Table 6) using Qwen2.5-72B as a representative strong open-source model; results for other LLMs exhibit consistent trends and are provided in Appendix B. HybridRAG-Bench reveals substantial and consistent performance gaps among KG-RAG baselines, far exceeding the gains obtained by scaling LLM size alone (Table 5). These gaps persist across domains and are especially pronounced when performance is analyzed by question type (Table 6), indicating that the benchmark meaningfully differentiates retrieval and reasoning strategies. Several clear and interpretable trends emerge in the breakdown by question type:

- **Multi-hop and hard multi-hop questions.** KG-RAG methods that explicitly model graph structure (e.g., ToG, ToG2.0, and EvoReasoner) consistently outperform text-only RAG. This reflects the importance of structured traversal and relational composition when reasoning requires chaining multiple facts.
- **Single-hop and conditional questions.** While these questions require limited multi-step composition, methods that effectively integrate both unstructured text and structured knowledge (e.g., EvoReasoner, and RAG+1-hop KG) generally achieves strong performance relative to text-only RAG and KG-only baselines. This suggests that even seemingly simple questions benefit from hybrid evidence integration, particularly when entity grounding or conditional constraints must be resolved.
- **Open-ended questions.** Methods with strong unstructured text retrieval perform better (e.g., RAG, RAG+1-hop KG, EvoReasoner), highlighting the importance of descriptive evidence and synthesis that cannot be fully captured by explicit relational connections alone.
- **Counterfactual questions.** Performance on counterfactual queries correlates strongly with a model's reasoning capability rather than retrieval effectiveness alone. Methods with explicit reasoning mechanisms (e.g., CoT, Self-Consistency, ToG2.0, and EvoReasoner) achieve substantially higher accuracy, while naïve KG augmentation (1-hop KG) performs very poorly. Inspection of model outputs shows that one-hop KG often induces overly cautious responses (e.g., "I don't know"), as local graph neighborhoods provide insufficient support for hypothetical reasoning. This highlights that counterfactual questions primarily test reasoning and uncertainty handling, rather than factual retrieval alone.

These results indicate that HybridRAG-Bench not only measures overall accuracy, but also diagnoses how different retrieval and reasoning mechanisms succeed or fail across distinct reasoning regimes. By disentangling performance across reasoning regimes, HybridRAG-Bench provides fine-grained diagnostic signals for evaluating KG-RAG systems.

5.3 KG Construction Effectiveness

Since HybridRAG-Bench relies on automatically constructed knowledge graphs, we evaluate whether our KG construction pipeline reliably recovers factual information from source documents. This analysis isolates the quality of KG extraction itself and does not involve downstream QA or reasoning performance.

Table 7: KG construction quality comparison.

Methods	OpenIE	GraphRAG	KGGen	EvoKG (Ours)
Facts Captured (Average on the 106 corpus)	29.36%	47.08%	66.46%	**71.36%**

We measure *fact recovery rate*, defined as the fraction of verifiable facts from the source corpus that are successfully recovered in the constructed KG:

$$\text{Recovery Rate} = \frac{\text{Number of facts recovered from corpus}}{\text{Number of verifiable facts in corpus}}.$$

We conduct this evaluation on the MINE benchmark [35], which contains 106 document corpora, each annotated with 15 manually verified facts. A fact is considered recovered if it can be matched to at least one triplet in the constructed KG.

Table 7 compares our pipeline with OpenIE, GraphRAG, and the state-of-the-art KGGen method. Our approach achieves the highest recovery rate, recovering approximately 71% of verifiable facts, outperforming KGGen by about 5%.

These results indicate that the KG construction component of HybridRAG-Bench is effective at extracting factual structure from text and is competitive with prior state-of-the-art methods. Importantly, this suggests that the difficulty of HybridRAG-Bench does not stem from weak or incomplete knowledge graph construction, but rather from the retrieval and reasoning demands imposed by the benchmark.

5.4 KG Construction Cost and Scalability

Since HybridRAG-Bench relies on automated KG construction, we assess the practical cost and scalability of the KG construction pipeline. Empirically, both extraction latency and token usage grow approximately linearly with document length, indicating that the main cost is dominated by LLM-based extraction over the input text. No superlinear cost growth is observed. In practice, KG construction is performed independently for each document and can therefore be parallelized across documents, substantially reducing end-to-end wall-clock time. Detailed latency and token scaling results are provided in Appendix A.

Token Cost Analysis. Figure 2 presents token usage as a function of document length during knowledge graph construction. As shown in the figure, token consumption increases approximately linearly with the size of the input corpus. This behavior arises from two primary factors: (1) input tokens scale proportionally with the raw text length, and (2) output tokens scale smoothly with the semantic density of each document. Because EvoKG performs a fixed and deterministic number of LLM extraction calls per document, the overall token footprint exhibits a consistent slope rather than abrupt jumps. This linear scaling pattern confirms that the computational cost of constructing the KG remains predictable and bounded. In practice, this ensures that EvoKG can support long-term or continuously updated corpora without unexpected surges in token usage, making the system practical for real-world deployment where cost control and scalability are critical.

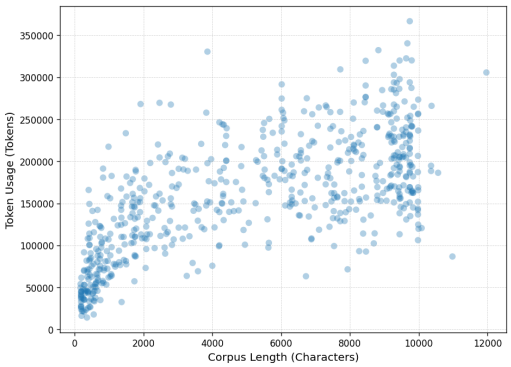


Figure 2: Token usage during KG construction plotted against corpus length. Token cost grows approximately linearly with input size due to proportional input tokens and a fixed number of extraction calls per document. The scaling pattern confirms that EvoKG's update cost remains predictable and stable across document sizes.

6 Conclusion

We presented HybridRAG-Bench, a benchmark construction framework for evaluating retrieval-intensive reasoning over hybrid knowledge. HybridRAG-Bench is designed to address long-standing challenges in evaluating LLM-based reasoning systems, particularly the difficulty of separating retrieval and reasoning from memorization when benchmarks overlap with model pretraining data. By constructing hybrid knowledge environments from recent scientific literature and generating questions grounded in explicit reasoning paths and supporting evidence, HybridRAG-Bench enables accurate evaluation of LLM-only, RAG-based, and hybrid KG-RAG methods. Our results show that the benchmark poses substantial challenges to LLM-only baselines and meaningfully distinguishes methods based on their ability to retrieve and integrate graph and textual information. While the framework relies on LLM-based components for knowledge extraction and question generation, and is currently instantiated using scientific literature, we view these choices as practical design decisions that enable scalable, contamination-aware evaluation rather than fundamental limitations. HybridRAG-Bench is intended as reusable research infrastructure, supporting flexible domain and temporal instantiation for future studies of retrieval, reasoning, and generalization over evolving knowledge.

Acknowledgments

This work is funded by the MIT-IBM AI Watson Lab, NSF awards #CCF-1845763, #CCF-2316235, and #CCF-2403237, a Google Faculty Research Award, and a Google Research Scholar Award.

References

[1] Aaron Adcock, Aayushi Srivastava, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pande, Abhinav Pandey, Abhinav Sharma, Abhishek Kadian, Abhishek Kumawat, Adam Kelsey, et al. 2026. The Llama 4 Herd: Architecture, Training, Evaluation, and Deployment Notes. *arXiv preprint arXiv:2601.11659* (2026).

[2] Akari Asai, Zeqiu Wu, Yizhong Wang, Avi Sil, and Hannaneh Hajishirzi. 2024. Self-rag: Learning to retrieve, generate, and critique through self-reflection. In *International Conference on Learning Representations (ICLR)*, Vol. 2024. 9112–9141.

Table 8: DeepSeek V3.2 performance (accuracy, %) grouped by question types. We highlight the first and second best results.

Datasets →	Arxiv-AI						Arxiv-CY						Arxiv-BIO					
Methods↓	Single-Hop	Single-Hop w. Conditions	Multi-Hop	Multi-Hop (Difficult)	Counterfactual	Open-Ended	Single-Hop	Single-Hop w. Conditions	Multi-Hop	Multi-Hop (Difficult)	Counterfactual	Open-Ended	Single-Hop	Single-Hop w. Conditions	Multi-Hop	Multi-Hop (Difficult)	Counterfactual	Open-Ended
IO	30.68±1.10	38.85±1.29	26.06±2.30	23.36±0.99	78.25±2.57	60.43±4.58	33.70±0.67	38.75±1.27	30.34±1.43	25.18±0.70	85.78±2.27	50.18±1.73	33.56±1.03	37.72±1.00	25.61±1.09	26.38±0.98	78.31±2.25	63.47±1.18
CoT	30.44±1.23	34.82±1.68	30.42±1.04	26.04±1.43	69.12±3.53	52.34±2.89	29.66±0.98	37.03±3.07	33.02±0.99	25.54±1.85	70.29±1.66	45.00±2.56	35.61±1.61	42.90±0.83	25.31±1.39	25.51±1.80	73.90±2.75	56.95±1.95
SC	29.40±1.20	35.83±2.00	29.70±2.60	24.30±1.55	66.67±3.88	45.53±2.89	28.99±1.13	35.94±2.61	31.14±1.83	21.69±1.90	65.20±1.52	41.25±2.49	35.91±1.60	40.93±1.70	19.80±7.08	24.78±2.88	64.07±2.49	51.47±3.43
RAG	48.33±0.19	68.11±0.90	42.22±0.57	27.07±0.63	16.37±3.94	70.92±4.37	47.69±0.21	69.53±0.00	41.28±0.34	25.30±0.00	11.85±1.45	70.09±2.23	50.57±0.19	69.43±0.52	32.91±1.28	19.93±0.36	15.25±0.00	72.11±1.05
1-hop KG	37.27±0.69	35.83±0.54	23.15±0.59	18.52±1.57	4.39±0.55	51.91±2.55	51.76±0.67	37.34±0.58	21.61±0.99	16.02±0.30	1.16±0.37	48.57±1.34	46.82±1.03	43.11±1.92	30.20±1.38	27.39±1.33	2.71±1.73	61.89±0.79
RAG + 1-hop KG	59.84±0.80	70.86±0.36	45.76±0.30	35.23±1.01	9.21±0.44	86.17±1.06	62.82±1.47	68.75±0.00	43.62±0.00	32.83±0.30	7.80±0.29	74.55±0.45	64.20±0.19	74.09±0.00	43.11±0.77	38.04±0.36	20.34±1.69	80.00±0.53
CoK	24.02±0.30	27.77±1.17	21.82±2.17	15.03±1.78	24.39±2.85	31.06±3.18	28.99±1.93	32.66±3.14	24.30±1.82	17.47±3.47	30.40±1.13	30.71±2.44	33.48±1.74	39.69±1.66	19.80±1.92	19.42±3.19	29.15±3.29	44.95±1.78
RoG	36.55±1.34	35.97±1.02	28.73±1.06	28.99±1.55	42.46±2.05	42.98±2.48	44.45±0.62	42.03±0.58	32.62±1.24	19.28±0.93	59.19±1.35	46.07±0.91	40.91±1.12	36.68±1.11	28.37±0.83	26.23±0.54	30.17±4.47	50.53±0.88
ToG	48.67±1.00	47.05±1.33	34.30±2.19	41.61±0.95	22.63±2.11	42.98±1.59	56.55±1.02	49.84±1.04	29.80±0.91	35.54±1.37	14.10±0.94	48.04±2.96	47.54±1.43	46.37±0.45	35.59±0.56	38.59±1.73	30.08±1.41	54.87±1.01
ToG2.0	62.01±1.93	60.29±1.90	47.52±1.25	53.56±0.99	96.32±1.02	90.64±3.46	69.24±1.04	64.69±0.91	44.70±0.91	41.93±1.77	97.11±0.97	78.04±1.45	61.44±0.94	69.43±1.70	44.59±0.89	51.88±2.18	95.59±1.36	84.84±0.91
PoG	48.84±0.83	45.61±0.98	33.33±1.27	40.40±1.15	13.68±1.31	38.72±3.66	59.33±0.81	46.09±1.10	30.20±1.47	31.20±0.70	5.90±0.99	48.04±1.31	47.50±0.78	44.04±0.46	31.73±1.59	40.58±0.79	12.20±3.46	53.89±1.58
R2KG	47.15±1.15	43.88±1.88	45.82±2.56	45.10±1.96	68.42±1.24	50.64±3.66	53.61±0.43	46.09±2.96	36.64±2.42	27.47±2.63	39.88±1.42	50.36±2.56	54.02±1.72	49.22±2.54	40.71±1.18	37.83±1.06	68.81±2.54	58.63±1.55
HippoRAG2.0	56.39±0.20	73.38±0.00	53.94±0.38	38.26±0.00	94.74±0.00	89.36±0.00	60.50±0.00	74.22±0.00	44.30±0.00	32.89±0.30	66.01±0.23	82.14±0.00	55.23±0.15	65.60±0.25	45.61±0.25	47.83±0.00	94.92±0.00	85.79±0.00
EvoReasoner	73.80±0.33	83.27±0.78	71.06±2.20	61.74±1.57	88.16±1.32	93.62±2.61	81.37±1.05	79.43±0.97	62.19±0.84	63.05±1.02	47.59±2.88	86.31±0.84	76.52±1.29	81.97±0.83	60.20±0.85	59.57±2.65	90.51±0.83	92.53±1.47

Table 9: Qwen2.5-72B performance (accuracy, %) grouped by question types. We highlight the first and second best results.

Datasets →	Arxiv-AI						Arxiv-CY						Arxiv-BIO					
Methods↓	Single-Hop	Single-Hop w. Conditions	Multi-Hop	Multi-Hop (Difficult)	Counterfactual	Open-Ended	Single-Hop	Single-Hop w. Conditions	Multi-Hop	Multi-Hop (Difficult)	Counterfactual	Open-Ended	Single-Hop	Single-Hop w. Conditions	Multi-Hop	Multi-Hop (Difficult)	Counterfactual	Open-Ended
IO	21.20±0.47	20.29±0.29	16.36±0.77	17.72±0.33	68.07±1.19	45.96±1.70	17.96±0.18	16.02±0.39	15.10±0.34	15.36±0.67	68.64±0.25	34.15±0.74	19.70±0.24	19.38±0.25	7.35±0.25	16.81±0.29	56.27±1.27	40.53±0.00
CoT	25.22±0.30	29.35±0.54	25.94±0.59	23.76±0.68	87.02±1.02	50.64±2.48	27.48±0.73	27.34±0.86	24.83±0.42	20.96±0.70	91.91±1.32	40.71±0.91	32.20±0.41	35.34±0.51	18.06±0.69	22.61±0.54	83.39±1.98	57.26±0.70
SC	25.46±1.21	26.76±1.67	24.00±0.82	22.68±1.30	84.04±0.66	50.64±2.08	26.30±0.73	27.19±1.74	23.09±1.51	21.45±1.97	87.05±3.22	40.54±0.91	29.55±0.63	33.47±1.69	19.29±1.09	21.01±1.30	83.05±3.03	56.42±0.70
RAG	53.55±0.19	67.63±0.59	41.82±0.49	29.31±0.32	48.54±0.41	85.11±0.00	51.68±0.00	68.36±0.39	35.23±1.01	23.49±0.00	23.41±0.29	72.32±0.00	51.70±0.19	68.91±0.00	34.69±0.00	22.10±0.36	40.68±0.00	79.21±0.26
1-hop KG	33.98±0.20	23.88±0.29	17.58±0.00	14.90±0.27	8.07±0.35	37.45±1.70	43.03±0.21	24.22±0.00	12.62±1.96	10.36±0.59	1.73±0.37	34.64±0.67	35.61±0.24	20.52±0.25	17.35±0.56	17.54±0.54	0.00±0.00	36.74±0.21
RAG + 1-hop KG	58.77±0.68	65.47±0.00	42.22±1.03	32.21±0.55	37.72±0.72	82.98±0.00	65.97±0.00	68.75±0.00	38.03±0.32	28.51±0.28	15.03±0.00	74.70±0.42	61.36±0.38	69.17±0.26	35.20±0.51	37.68±0.00	35.59±0.00	82.63±0.00
CoK	37.19±1.60	41.01±2.13	33.21±1.50	31.81±2.90	74.74±2.68	66.81±3.95	34.45±2.04	35.00±3.22	28.19±2.68	23.61±2.10	69.83±1.73	55.18±2.61	42.58±1.69	43.63±2.44	25.51±2.16	29.71±1.30	71.53±4.72	62.00±1.71
RoG	37.43±0.53	32.66±1.08	27.64±0.62	30.87±0.42	71.23±2.44	33.62±0.85	47.98±0.81	32.81±0.99	25.77±0.81	18.19±0.96	71.45±1.07	41.79±1.04	38.79±3.53	38.03±0.41	22.86±1.27	29.13±0.54	77.97±3.39	50.21±0.86
ToG	53.09±0.39	35.11±0.70	38.79±1.38	39.60±0.74	45.09±1.31	60.85±1.70	57.14±0.88	40.31±1.61	29.53±1.20	25.78±1.50	13.99±0.85	55.00±0.91	45.83±0.71	40.28±0.56	32.78±0.75	45.29±0.36	45.34±2.20	58.68±0.26
ToG2.0	59.52±0.30	54.82±0.70	44.73±0.45	43.89±1.32	90.35±1.24	83.40±1.59	63.61±0.57	55.78±1.27	29.53±1.20	29.16±1.64	96.07±0.43	69.46±1.04	60.08±0.66	58.24±0.70	38.98±0.83	51.88±0.58	93.56±1.27	80.84±0.42
PoG	51.65±0.20	36.12±0.54	30.55±1.56	35.70±0.78	53.68±0.35	60.85±1.70	54.62±0.46	38.91±0.58	28.19±0.95	19.40±0.59	20.92±0.85	50.89±0.98	46.21±0.63	38.13±0.78	29.80±0.41	42.17±0.54	60.00±1.36	58.42±0.74
R2KG	43.13±0.87	37.99±1.06	36.36±1.63	30.34±0.89	73.68±2.60	45.53±1.70	50.34±1.08	42.50±1.61	30.87±0.42	26.14±0.30	60.92±1.25	48.39±1.43	50.53±0.51	39.38±0.73	35.31±0.59	32.32±4.31	64.07±3.28	53.37±1.08
HippoRAG2.0	17.67±1.81	29.78±3.74	20.48±3.88	5.23±0.27	23.68±1.75	30.21±7.66	13.45±0.00	19.53±0.00	8.59±1.61	4.58±1.69	14.34±1.39	14.29±0.00	30.30±0.24	41.45±0.00	22.45±0.00	11.88±0.35	56.27±0.68	46.11±0.26
EvoReasoner	71.75±0.76	75.54±0.59	65.86±3.02	53.47±2.70	88.30±4.20	90.78±4.37	69.75±3.40	63.28±31.65	54.50±1.23	49.04±1.46	83.12±1.23	82.86±2.07	75.15±0.78	77.82±1.41	56.94±1.95	56.09±2.03	91.53±1.07	89.89±0.61

Table 10: LLaMa 3.3-70B performance (accuracy, %) grouped by question types. We highlight the first and second best results.

Datasets →	Arxiv-AI						Arxiv-CY						Arxiv-BIO					
Methods↓	Single-Hop	Single-Hop w. Conditions	Multi-Hop	Multi-Hop (Difficult)	Counterfactual	Open-Ended	Single-Hop	Single-Hop w. Conditions	Multi-Hop	Multi-Hop (Difficult)	Counterfactual	Open-Ended	Single-Hop	Single-Hop w. Conditions	Multi-Hop	Multi-Hop (Difficult)	Counterfactual	Open-Ended
IO	31.93±0.20	31.83±0.31	22.58±0.50	20.30±0.87	73.03±1.14	46.81±2.13	27.39±0.49	31.09±0.91	29.93±0.68	15.90±0.30	71.56±1.18	36.96±0.71	29.36±0.33	32.90±0.26	20.28±0.66	25.18±0.60	77.54±0.73	51.97±1.14
CoT	32.73±0.88	33.99±0.60	25.91±1.16	20.30±0.87	75.66±1.14	39.89±1.76	29.58±0.43	33.44±1.04	30.60±1.62	27.11±1.01	80.12±1.07	44.82±0.36	34.09±0.86	36.68±0.83	25.20±1.10	28.41±0.71	74.24±1.98	50.74±1.55
SC	35.14±1.19	33.63±0.93	27.12±1.38	18.96±0.87	76.75±4.04	39.36±3.84	30.42±0.57	33.59±0.70	33.56±0.95	26.75±0.90	79.31±1.23	45.89±2.00	33.41±0.88	38.45±1.20	24.49±0.97	26.38±0.74	73.56±1.73	50.84±1.51
RAG	51.61±0.45	66.73±0.31	39.24±0.50	18.96±0.87	47.37±0.62	67.02±1.84	50.28±0.52	66.67±0.37	37.36±0.32	27.31±0.28	19.27±0.72	67.56±0.42	47.73±0.76	65.80±0.00	35.71±0.51	25.00±0.36	44.07±0.00	67.37±0.00
1-hop KG	32.33±0.35	31.47±0.60	27.27±0.43	21.14±0.34	9.65±0.00	32.45±0.92	41.76±0.43	29.38±0.62	23.36±0.27	14.34±0.45	3.24±0.28	22.86±0.71	37.35±0.39	31.30±0.41	26.63±0.20	20.43±0.54	8.47±0.00	34.21±0.58
RAG + 1-hop KG	59.64±0.60	66.19±0.00	46.97±0.30	32.21±0.00	39.91±0.44	71.28±1.06	59.80±3.07	63.28±6.63	40.94±2.85	27.71±0.98	14.07±1.66	68.15±3.37	61.55±0.19	67.88±0.00	39.29±0.00	30.43±0.00	43.22±0.85	68.16±0.26
CoK	34.94±1.22	36.26±1.33	29.45±1.46	21.14±0.34	74.56±1.66	48.51±4.34	31.18±1.17	32.03±0.86	33.15±2.06	23.01±1.39	72.14±1.29	43.93±1.54	34.55±0.44	43.73±1.02	26.22±1.78	24.20±1.87	74.58±4.01	58.21±2.17
RoG	41.16±0.45	39.03±0.60	34.24±0.68	36.91±0.47	76.97±0.96	40.43±1.50	49.41±1.21	39.06±0.00	36.78±0.50	24.70±1.26	66.59±0.99	43.57±0.67	42.95±0.45	39.69±0.70	29.80±0.25	25.80±0.98	65.08±1.73	49.37±0.61
ToG	50.80±0.35	42.27±1.28	36.97±0.74	42.28±0.47	10.53±1.39	51.06±1.50	53.28±0.62	41.25±0.31	32.89±1.85	33.13±1.37	2.43±0.43	41.25±1.04	50.38±0.76	44.56±0.73	35.92±0.83	34.20±1.16	11.19±1.36	50.53±1.25
ToG2.0	59.44±0.40	55.22±0.93	46.67±0.96	43.46±1.20	92.54±2.01	85.11±1.50	64.37±0.90	62.03±0.62	37.85±1.73	34.22±0.45	93.29±0.69	71.96±1.66	35.91±5.48	42.38±3.48	41.02±1.10	48.99±1.08	91.53±1.86	81.47±0.52
PoG	49.70±0.66	41.55±0.31	39.39±0.74	41.78±0.56	11.18±0.73	47.87±1.84	53.28±0.56	40.31±0.38	32.48±1.24	31.81±0.59	2.54±0.28	39.82±0.91	49.55±0.15	44.46±0.51	34.18±0.32	32.75±1.41	12.88±0.83	51.68±0.39
R2KG	48.90±0.77	46.04±0.51	38.94±0.50	40.44±1.45	63.38±2.73	39.89±1.76	48.74±1.16	51.56±1.91	42.82±2.18	36.02±0.80	48.55±0.73	49.46±1.84	49.77±0.66	52.64±1.45	38.78±1.12	32.03±0.54	70.51±2.30	58.95±1.60
HippoRAG2.0	52.01±0.67	66.01±0.93	48.33±0.26	30.87±0.00	87.72±0.00	82.45±1.76	47.14±0.62	71.09±3.16	40.40±0.78	27.83±0.96	87.28±0.63	71.79±0.91	50.98±0.19	64.77±0.00	41.12±0.25	33.77±0.58	85.42±1.36	80.21±0.26
EvoReasoner	71.29±0.20	76.98±0.72	63.64±1.21	53.36±1.01	56.14±0.88	85.11±0.00	75.29±1.60	77.81±2.07	57.05±1.53	51.08±1.76	56.76±1.43	83.75±1.54	75.57±1.43	76.94±0.45	53.95±0.91	53.44±0.94	53.81±1.85	89.34±1.31

[3] Shuai Bai, Yuxuan Cai, Ruizhe Chen, Keqin Chen, Xionghui Chen, Zesen Cheng, Lianghao Deng, Wei Ding, Chang Gao, Chunjiang Ge, et al. 2025. Qwen3-VL technical report. *arXiv preprint arXiv:2511.21631* (2025).

Table 11: LLaMa 3.1-8B performance (accuracy, %) grouped by question types. We highlight the first and second best results.

Datasets →	Arxiv-AI						Arxiv-CY						Arxiv-BIO					
Methods↓	Single-Hop	Single-Hop w. Conditions	Multi-Hop	Multi-Hop (Difficult)	Counterfactual	Open-Ended	Single-Hop	Single-Hop w. Conditions	Multi-Hop	Multi-Hop (Difficult)	Counterfactual	Open-Ended	Single-Hop	Single-Hop w. Conditions	Multi-Hop	Multi-Hop (Difficult)	Counterfactual	Open-Ended
IO	$20.48_{\pm0.92}$	$19.57_{\pm0.54}$	$12.48_{\pm1.12}$	$14.90_{\pm1.49}$	$56.67_{\pm1.53}$	$26.38_{\pm2.17}$	$16.89_{\pm0.49}$	$16.25_{\pm0.58}$	$16.24_{\pm0.27}$	$12.05_{\pm0.85}$	$54.22_{\pm1.81}$	$29.82_{\pm1.07}$	$19.85_{\pm0.62}$	$26.11_{\pm0.96}$	$11.73_{\pm0.72}$	$17.39_{\pm0.46}$	$57.29_{\pm1.66}$	$40.95_{\pm1.50}$
CoT	$25.54_{\pm0.32}$	$27.34_{\pm1.93}$	$20.85_{\pm0.73}$	$17.32_{\pm0.99}$	$54.56_{\pm1.95}$	$31.06_{\pm3.18}$	$25.71_{\pm1.17}$	$21.25_{\pm1.51}$	$24.03_{\pm1.30}$	$20.36_{\pm1.34}$	$50.06_{\pm2.98}$	$27.14_{\pm0.44}$	$26.36_{\pm0.98}$	$31.19_{\pm3.76}$	$17.86_{\pm0.91}$	$21.16_{\pm1.80}$	$51.53_{\pm2.03}$	$44.21_{\pm2.47}$
SC	$26.59_{\pm2.15}$	$25.76_{\pm0.29}$	$19.76_{\pm1.94}$	$16.24_{\pm1.23}$	$53.86_{\pm4.10}$	$26.38_{\pm1.70}$	$25.38_{\pm1.24}$	$24.06_{\pm1.88}$	$26.31_{\pm1.77}$	$20.24_{\pm1.50}$	$54.57_{\pm2.27}$	$25.89_{\pm1.49}$	$25.68_{\pm1.98}$	$31.71_{\pm0.39}$	$18.57_{\pm1.35}$	$22.17_{\pm3.06}$	$57.97_{\pm2.92}$	$39.47_{\pm2.35}$
RAG	$47.59_{\pm0.20}$	$60.07_{\pm1.80}$	$34.85_{\pm0.30}$	$23.15_{\pm0.34}$	$3.51_{\pm0.00}$	$68.09_{\pm0.00}$	$47.69_{\pm0.21}$	$63.28_{\pm0.00}$	$31.88_{\pm1.01}$	$22.29_{\pm0.00}$	$2.31_{\pm0.00}$	$54.02_{\pm0.45}$	$47.35_{\pm0.00}$	$64.77_{\pm0.00}$	$30.10_{\pm0.51}$	$25.00_{\pm0.36}$	$1.69_{\pm0.00}$	$58.68_{\pm0.79}$
1-hop KG	$32.29_{\pm0.32}$	$25.32_{\pm0.54}$	$20.12_{\pm0.89}$	$18.52_{\pm1.73}$	$0.88_{\pm0.00}$	$37.02_{\pm1.70}$	$45.97_{\pm0.68}$	$26.56_{\pm1.10}$	$16.91_{\pm1.15}$	$11.08_{\pm0.61}$	$0.00_{\pm0.00}$	$36.25_{\pm1.66}$	$35.61_{\pm0.34}$	$28.08_{\pm0.21}$	$20.71_{\pm0.83}$	$17.10_{\pm1.69}$	$0.00_{\pm0.00}$	$45.68_{\pm0.84}$
RAG + 1-hop KG	**$53.82_{\pm0.40}$**	**$68.35_{\pm0.72}$**	$40.91_{\pm0.30}$	$29.53_{\pm0.67}$	$3.51_{\pm0.00}$	$70.21_{\pm0.00}$	**$59.87_{\pm0.21}$**	**$69.53_{\pm0.78}$**	$33.56_{\pm0.00}$	$28.92_{\pm0.60}$	$1.73_{\pm0.00}$	$70.09_{\pm0.45}$	**$57.77_{\pm0.57}$**	**$68.65_{\pm0.26}$**	$32.91_{\pm0.77}$	$34.06_{\pm0.72}$	$5.08_{\pm1.69}$	$70.00_{\pm0.53}$
CoK	$29.64_{\pm5.72}$	$33.67_{\pm2.96}$	$25.09_{\pm3.03}$	$23.22_{\pm2.19}$	$76.84_{\pm3.95}$	$49.36_{\pm4.34}$	$30.25_{\pm2.33}$	$31.56_{\pm2.60}$	$25.50_{\pm1.12}$	$15.90_{\pm1.73}$	$76.18_{\pm1.57}$	$42.68_{\pm3.72}$	$34.09_{\pm2.57}$	$35.75_{\pm2.60}$	$21.02_{\pm2.24}$	$23.91_{\pm2.25}$	$67.12_{\pm4.87}$	$53.37_{\pm1.40}$
RoG	$28.59_{\pm1.06}$	$22.45_{\pm0.54}$	$16.12_{\pm1.41}$	$22.42_{\pm1.83}$	$42.28_{\pm1.70}$	$23.40_{\pm1.35}$	$32.77_{\pm1.68}$	$24.22_{\pm1.48}$	$17.18_{\pm2.19}$	$10.24_{\pm1.48}$	$41.73_{\pm1.69}$	$25.71_{\pm2.96}$	$28.64_{\pm1.65}$	$24.46_{\pm1.52}$	$13.78_{\pm1.02}$	$8.12_{\pm1.06}$	$39.32_{\pm2.92}$	$35.58_{\pm1.65}$
ToG	$19.44_{\pm1.21}$	$23.74_{\pm2.03}$	$13.09_{\pm1.31}$	$20.13_{\pm3.03}$	$1.75_{\pm1.24}$	$34.89_{\pm2.89}$	$24.37_{\pm5.50}$	$19.14_{\pm5.11}$	$12.08_{\pm3.36}$	$9.19_{\pm1.56}$	$0.87_{\pm0.65}$	$30.13_{\pm1.93}$	$16.89_{\pm6.30}$	$15.85_{\pm4.33}$	$11.53_{\pm4.00}$	$12.90_{\pm4.45}$	$0.68_{\pm0.83}$	$26.84_{\pm9.28}$
ToG2.0	$52.45_{\pm1.00}$	$45.18_{\pm0.84}$	$30.06_{\pm1.06}$	**$41.07_{\pm2.14}$**	$77.02_{\pm1.87}$	$70.21_{\pm3.01}$	$56.62_{\pm0.91}$	$53.91_{\pm1.91}$	$29.36_{\pm2.34}$	$31.17_{\pm1.16}$	**$79.91_{\pm1.93}$**	$57.14_{\pm2.75}$	$50.53_{\pm4.72}$	$52.23_{\pm1.20}$	$37.04_{\pm0.52}$	**$35.07_{\pm1.63}$**	$74.58_{\pm2.14}$	$72.21_{\pm1.87}$
PoG	$17.67_{\pm1.68}$	$17.70_{\pm1.08}$	$15.27_{\pm0.80}$	$20.67_{\pm1.37}$	$2.46_{\pm0.35}$	$32.77_{\pm3.18}$	$24.54_{\pm3.07}$	$15.31_{\pm4.27}$	$11.14_{\pm2.27}$	$10.12_{\pm0.89}$	$1.39_{\pm0.28}$	$34.46_{\pm1.66}$	$20.23_{\pm2.25}$	$17.93_{\pm3.51}$	$18.67_{\pm1.76}$	$22.17_{\pm1.63}$	$0.68_{\pm0.83}$	$34.32_{\pm2.73}$
R2KG	$26.75_{\pm1.52}$	$26.91_{\pm1.08}$	$20.73_{\pm2.11}$	$23.22_{\pm2.31}$	$34.04_{\pm3.82}$	$25.96_{\pm5.93}$	$36.97_{\pm2.14}$	$26.56_{\pm2.84}$	$22.28_{\pm1.77}$	$16.75_{\pm1.54}$	$30.87_{\pm4.50}$	$28.75_{\pm2.49}$	$31.29_{\pm1.47}$	$25.80_{\pm1.84}$	$18.37_{\pm1.96}$	$19.28_{\pm3.96}$	$30.51_{\pm6.95}$	$39.26_{\pm1.13}$
HippoRAG2.0	$43.37_{\pm0.00}$	$55.40_{\pm0.00}$	$40.85_{\pm0.30}$	$24.43_{\pm0.33}$	**$81.05_{\pm0.43}$**	$67.66_{\pm1.59}$	$43.36_{\pm0.31}$	$56.72_{\pm0.38}$	$36.51_{\pm0.33}$	$23.49_{\pm0.38}$	$73.29_{\pm0.43}$	$68.21_{\pm0.44}$	$43.41_{\pm0.19}$	$64.87_{\pm0.39}$	$32.14_{\pm0.00}$	$32.90_{\pm0.74}$	**$76.27_{\pm0.00}$**	$68.53_{\pm0.21}$
EvoReasoner	$53.73_{\pm0.59}$	$62.16_{\pm0.98}$	**$49.58_{\pm2.28}$**	$35.84_{\pm2.90}$	$78.42_{\pm2.39}$	**$77.02_{\pm4.13}$**	$55.97_{\pm0.67}$	$60.94_{\pm1.78}$	**$40.27_{\pm1.12}$**	**$35.06_{\pm1.39}$**	$70.75_{\pm2.52}$	**$72.86_{\pm2.86}$**	$56.29_{\pm0.98}$	$66.42_{\pm1.29}$	**$42.55_{\pm1.63}$**	$34.35_{\pm1.63}$	$74.24_{\pm4.21}$	**$80.63_{\pm0.91}$**

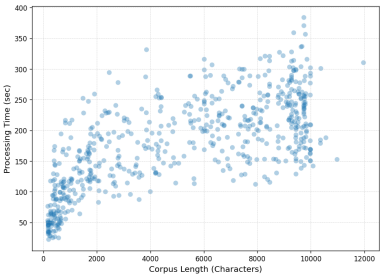


Figure 3: Per-document KG construction latency as a function of corpus length (in characters). Longer documents incur higher extraction time, and the trend follows the expected near-linear scaling dominated by LLM token processing.

[4] Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. 2013. Semantic parsing on Freebase from question-answer pairs. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing (EMNLP)*. 1533–1544.

[5] Kurt Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the ACM SIGMOD International Conference on Management of Data*. 1247–1250.

[6] Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language Models Are Few-Shot Learners. *Advances in Neural Information Processing Systems (NeurIPS)* 33 (2020), 1877–1901.

[7] Shulin Cao, Jiaxin Shi, Liangming Pan, Lunyiu Nie, Yutong Xiang, Lei Hou, Juanzi Li, Bin He, and Hanwang Zhang. 2022. KQA pro: A dataset with explicit compositional programs for complex question answering over knowledge base. In *Proceedings of the 60th annual meeting of the Association for Computational Linguistics (volume 1: long papers)*. 6101–6119.

[8] Liyi Chen, Panrong Tong, Zhongming Jin, Ying Sun, Jieping Ye, and Hui Xiong. 2024. Plan-on-Graph: Self-Correcting Adaptive Planning of Large Language Model on Knowledge Graphs. In *Advances in neural information processing systems (NeurIPS)*.

[9] Ziyang Chen, Jinzhi Liao, and Xiang Zhao. 2023. Multi-granularity temporal question answering over knowledge graphs. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 11378–11392.

[10] Wei-Lin Chiang, Lianmin Zheng, Ying Sheng, Anastasios Nikolas Angelopoulos, Tianle Li, Dacheng Li, Banghua Zhu, Hao Zhang, Michael Jordan, Joseph E Gonzalez, et al. 2024. Chatbot arena: An open platform for evaluating LLMs by human preference. In *International Conference on Machine Learning (ICML)*.

[11] Preetam Prabhu Srikar Dammu, Himanshu Naidu, and Chirag Shah. 2025. Dynamic-kgqa: A scalable framework for generating adaptive question answering datasets. In *Proceedings of the 48th International ACM SIGIR Conference on Research and Development in Information Retrieval*. 3498–3508.

[12] Yihong Dong, Xue Jiang, Huanyu Liu, Zhi Jin, Bin Gu, Mengfei Yang, and Ge Li. 2024. Generalization or memorization: Data contamination and trustworthy evaluation for large language models. In *Findings of the Association for Computational Linguistics: ACL 2024*. 12039–12050.

[13] Mohnish Dubey, Debayan Banerjee, Abdelrahman Abdelkawi, and Jens Lehmann. 2019. LC-QuAD 2.0: A large dataset for complex question answering over Wikidata and DBpedia. In *International Semantic Web Conference*. 69–78.

[14] Darren Edge, Ha Trinh, Newman Cheng, Joshua Bradley, Alex Chao, Apurva Mody, Steven Truitt, Dasha Metropolitansky, Robert Osazuwa Ness, and Jonathan Larson. 2024. From local to global: A graph RAG approach to query-focused summarization. *arXiv preprint arXiv:2404.16130* (2024).

[15] Clémentine Fourrier, Nathan Habib, Alina Lozovskaya, Konrad Szafer, and Thomas Wolf. 2024. Open LLM Leaderboard v2. https://huggingface.co/spaces/open-llm-leaderboard/open_llm_leaderboard.

[16] Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, et al. 2024. The Llama 3 herd of models. *arXiv preprint arXiv:2407.21783* (2024).

[17] Jiawei Gu, Xuhui Jiang, Zhichao Shi, Hexiang Tan, Xuehao Zhai, Chengjin Xu, Wei Li, Yinghan Shen, Shengjie Ma, Honghao Liu, et al. 2026. A survey on LLM-as-a-Judge. *The Innovation* (2026), 101253.

[18] Yu Gu, Sue Kase, Michelle Vanni, Brian Sadler, Percy Liang, Xifeng Yan, and Yu Su. 2021. Beyond I.I.D.: three levels of generalization for question answering on knowledge bases. In *Proceedings of the Web Conference*. 3477–3488.

[19] Zirui Guo, Lianghao Xia, Yanhua Yu, Tu Ao, and Chao Huang. 2025. LightRAG: Simple and Fast Retrieval-Augmented Generation. In *Findings of the Association for Computational Linguistics: EMNLP*. 10746–10761.

[20] Bernal Jiménez Gutiérrez, Yiheng Shu, Weijian Qi, Sizhe Zhou, and Yu Su. 2025. From RAG to Memory: Non-Parametric Continual Learning for Large Language Models. In *International Conference on Machine Learning (ICML)*.

[21] Zhen Jia, Soumajit Pramanik, Rishiraj Saha Roy, and Gerhard Weikum. 2021. Complex temporal question answering on knowledge graphs. In *Proceedings of the 30th ACM International Conference on Information & Knowledge Management*. 792–802.

[22] Sumin Jo, Junseong Choi, Jiho Kim, and Edward Choi. 2025. R2-kg: General-purpose dual-agent framework for reliable reasoning on knowledge graphs. In *Proceedings of the 14th International Joint Conference on Natural Language Processing and the 4th Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics*. 486–509.

[23] Mandar Joshi, Eunsol Choi, Daniel S Weld, and Luke Zettlemoyer. 2017. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 1601–1611.

[24] Jens Lehmann, Robert Isele, Max Jakob, Anja Jentzsch, Dimitris Kontokostas, Pablo N Mendes, Sebastian Hellmann, Mohamed Morsey, Patrick Van Kleef, Sören Auer, et al. 2015. DBpedia–a large-scale, multilingual knowledge base extracted from Wikipedia. *Semantic Web* 6, 2 (2015), 167–195.

[25] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive NLP tasks. *Advances in Neural Information Processing Systems (NeurIPS)* 33 (2020), 9459–9474.

[26] Xingxuan Li, Ruochen Zhao, Yew Ken Chia, Bosheng Ding, Shafiq Joty, Soujanya Poria, and Lidong Bing. 2024. Chain-of-Knowledge: Grounding Large Language Models via Dynamic Knowledge Adapting over Heterogeneous Sources. In *International Conference on Learning Representations (ICLR)*.

[27] Yucheng Li, Yunhao Guo, Frank Guerin, and Chenghua Lin. 2024. An open-source data contamination report for large language models. In *Findings of the Association for Computational Linguistics: EMNLP*. 528–541.

[28] Yanhong Li, Tianyang Xu, Kenan Tang, Karen Livescu, David McAllester, and Jiawei Zhou. 2025. OKBench: Democratizing LLM Evaluation with Fully Automated, On-Demand, Open Knowledge Benchmarking. *arXiv preprint arXiv:2511.08598* (2025).

[29] Zi Liang, Liantong Yu, Zhang Shiyu, Qingqing Ye, and Haibo Hu. 2026. How Much Do Large Language Model Cheat on Evaluation? Benchmarking Overestimation under the One-Time-Pad-Based Framework. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 40. 37636–37644.

[30] Junhong Lin, Song Wang, Xiaojie Guo, Julian Shun, and Yada Zhu. 2025. Temporal reasoning with large language models augmented by evolving knowledge graphs. *arXiv preprint arXiv:2509.15464* (2025).

[31] Aixin Liu, Aoxue Mei, Bangcai Lin, Bing Xue, Bingxuan Wang, Bingzheng Xu, Bochao Wu, Bowei Zhang, Chaofan Lin, Chen Dong, et al. 2025. DeepSeek-V3.2: Pushing the frontier of open large language models. *arXiv preprint arXiv:2512.02556* (2025).

[32] Linhao Luo, Yuan-Fang Li, Reza Haf, and Shirui Pan. 2024. Reasoning on Graphs: Faithful and Interpretable Large Language Model Reasoning. In *International Conference on Learning Representations (ICLR)*.

[33] Shengjie Ma, Chengjin Xu, Xuhui Jiang, Muzhi Li, Huaren Qu, Cehao Yang, Jiaxin Mao, and Jian Guo. 2025. Think-on-Graph 2.0: Deep and Faithful Large Language Model Reasoning with Knowledge-guided Retrieval Augmented Generation. In *International Conference on Learning Representations (ICLR)*.

[34] Yu. A. Malkov and Dmitry A. Yashunin. 2018. Efficient and robust approximate nearest neighbor search using hierarchical navigable small world graphs. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 42, 4 (2018), 824–836.

[35] Belinda Mo, Kyssen Yu, Joshua Kazdan, Proud Mpala, Lisa Yu, Chris Cundy, Charilaos Kanatsoulis, and Sanmi Koyejo. 2025. KGGen: Extracting Knowledge Graphs from Plain Text with Language Models. *Advances in Neural Information Processing Systems (NeurIPS)* (2025).

[36] Medha Palavalli, Amanda Bertsch, and Matthew R Gormley. 2024. A taxonomy for data contamination in large language models. In *Proceedings of the 1st Workshop on Data Contamination (CONDA)*. 22–40.

[37] Aaditya Singh, Adam Fry, Adam Perelman, Adam Tart, Adi Ganesh, Ahmed El-Kishky, Aidan McLaughlin, Aiden Low, AJ Ostrow, Akhila Ananthram, et al. 2025. OpenAI GPT-5 system card. *arXiv preprint arXiv:2601.03267* (2025).

[38] Aaditya K Singh, Muhammed Yusuf Kocyigit, Andrew Poulton, David Esiobu, Maria Lomeli, Gergely Szilvasy, and Dieuwke Hupkes. 2024. Evaluation data contamination in LLMs: how do we measure it and (when) does it matter? *arXiv preprint arXiv:2411.03923* (2024).

[39] Jiashuo Sun, Chengjin Xu, Lumingyuan Tang, Saizhuo Wang, Chen Lin, Yeyun Gong, Lionel Ni, Heung-Yeung Shum, and Jian Guo. 2024. Think-on-Graph: Deep and Responsible Reasoning of Large Language Model on Knowledge Graph. In *International Conference on Learning Representations (ICLR)*.

[40] Alon Talmor and Jonathan Berant. 2018. The Web as a Knowledge-base for Answering Complex Questions. In *Proceedings of NAACL-HLT*. 641–651.

[41] Priyansh Trivedi, Gaurav Maheshwari, Mohnish Dubey, and Jens Lehmann. 2017. LC-QuAD: A corpus for complex question answering over knowledge graphs. In *International Semantic Web Conference*. 210–218.

[42] Denny Vrandečić and Markus Krötzsch. 2014. Wikidata: a free collaborative knowledgebase. *Commun. ACM* 57, 10 (2014), 78–85.

[43] Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc V Le, Ed H Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2023. Self-Consistency Improves Chain of Thought Reasoning in Language Models. In *International Conference on Learning Representations (ICLR)*.

[44] Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems (NeurIPS)* 35 (2022), 24824–24837.

[45] Cheng Xu, Shuhao Guan, Derek Greene, M Kechadi, et al. 2024. Benchmark data contamination of large language models: A survey. *arXiv preprint arXiv:2406.04244* (2024).

[46] An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxi Yang, Jingren Zhou, Junyang Lin, Kai Dang, Keming Lu, Keqin Bao, Kexin Yang, Le Yu, Mei Li, Mingfeng Xue, Pei Zhang, Qin Zhu, Rui Men, Runji Lin, Tianhao Li, Tingyu Xia, Xingzhang Ren, Xuancheng Ren, Yang Fan, Yang Su, Yichang Zhang, Yu Wan, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, and Zihan Qiu. 2024. Qwen2.5 Technical Report. *arXiv preprint arXiv:2412.15115* (2024).

[47] Xiao Yang, Kai Sun, Hao Xin, Yushi Sun, Nikita Bhalla, Xiangsen Chen, Sajal Choudhary, Rongze Gui, Ziran Jiang, Ziyu Jiang, et al. 2024. CRAG-comprehensive RAG benchmark. *Advances in Neural Information Processing Systems (NeurIPS)* 37 (2024), 10470–10490.

[48] Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D Manning. 2018. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. 2369–2380.

[49] Wen-tau Yih, Matthew Richardson, Christopher Meek, Ming-Wei Chang, and Jina Suh. 2016. The value of semantic parse labeling for knowledge base question answering. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*. 201–206.

[50] Liangliang Zhang, Zhuorui Jiang, Hongliang Chi, Haoyang Chen, Mohammed Elkoumy, Fali Wang, Qiong Wu, Zhengyi Zhou, Shirui Pan, Suhang Wang, et al. 2025. Diagnosing and Addressing Pitfalls in KG-RAG Datasets: Toward More Reliable Benchmarking. *Advances in Neural Information Processing Systems (NeurIPS)* (2025).

[51] Yuyu Zhang, Hanjun Dai, Zornitsa Kozareva, Alexander Smola, and Le Song. 2018. Variational reasoning for question answering with knowledge graph. In *Proceedings of the AAAI Conference on Artificial Intelligence*, Vol. 32.

A KG Construction Efficiency and Cost Analysis

Theoretical Complexity Analysis. Let n and m denote the number of extracted entities and relations from a corpus, and N and M denote the number of nodes and edges, respectively, in the existing KG. KG extraction requires a single LLM call with $O(n+m)$ cost. Entity and relation alignment are performed using HNSW-based similarity search [34], with expected complexity $O(n \log N + m \log M)$ in practice.[2] Subsequent merging and insertion operations involve lightweight attribute aggregation and incur $O(n+m)$ cost.

Overall, the complexity of KG evolution is $O(n \log N + m \log M)$. In addition, the pipeline uses a constant number of LLM calls per corpus (five in our implementation), making it practical for continuous and large-scale KG updates.

Empirical Complexity Analysis. To assess the efficiency of KG construction, we measure the per-document extraction time across all corpora. Figure 3 plots the KG construction latency as a function of document length (in characters). The results show that longer documents incur proportional higher processing time. This behavior is expected, as extraction latency is dominated by the LLM inference, whose computational cost grows with input length. Empirically, the latency grows with document size and exhibits no abrupt increase as corpus length increases. Variance in the vertical direction is attributable to LLM-provider latency: extraction is executed via the OpenRouter API, whose backend routing introduces heterogeneity in response times that is unrelated to algorithmic complexity.

B Additional Per-Question-Type Results.

To provide a more complete view of benchmark behavior across models, we report detailed per-question-type performance for all evaluated LLMs, including DeepSeek-V3.2 (Table 8), Qwen2.5-72B (Table 9), LLaMA-3.3 70B (Table 10), and LLaMA-3.1-8B (Table 11). For each model and baseline, we report accuracy and standard deviation across 5 runs, grouped by question type.

[2]The theoretical complexity bound is derived under the assumption of using exact Delaunay graphs with bounded average degree instead of approximate Delaunay graphs to construct the HNSW index [34]. Although this is not how the HNSW index is constructed in practice, the authors show that the bound seems to hold in simulations on low-dimensional data.