

# Towards Holistic Knowledge Retrieval for Augmented Generation: Integrating Graph-Grounded and Query-Aligned Knowledge

Weidong Zhang<sup>1</sup>, Zhe Wang<sup>1†</sup>, Kewen Wang<sup>1</sup>, Haifeng Guo<sup>2</sup>, Shuxuan Lian<sup>3</sup>, Weizhi Du<sup>4</sup>

<sup>1</sup>School of Information and Communication Technology, Griffith University, Australia

<sup>2</sup>School of Data Science, Lingnan University, Hong Kong

<sup>3</sup>Business School, University of New South Wales, Australia

<sup>4</sup>Department of Electrical Engineering and Computer Science, University of Michigan, USA

zhe.wang@griffith.edu.au

## Abstract

Retrieval-Augmented Generation (RAG) enables Large Language Models (LLMs) to access external knowledge during generation, improving factual reliability without parameter re-training. However, existing methods still struggle to retrieve evidence that is both relevant and complete. Standard RAG may miss semantic associations among related chunks. Graph-based RAG addresses this issue by modeling entity relations, but it is often query-agnostic. Question-format RAG further improves query matching by converting facts into Question-Answer (QA) pairs, but it cannot guarantee semantic association and introduces knowledge fragmentation. To address these issues, we propose **HolisticRAG**, a framework integrating graph-grounded and query-aligned knowledge for holistic knowledge RAG. In the offline stage, HolisticRAG constructs a knowledge representation to preserve semantic associations while reducing query-agnostic retrieval. In the online stage, it applies holistic knowledge clustering to merge all fragmented evidence and adaptively retrieve sufficient supporting facts. Experiments demonstrate the effectiveness of HolisticRAG<sup>1</sup>.

## 1 Introduction

Retrieval-Augmented Generation (RAG) enhances the capabilities of Large Language Models (LLMs) by retrieving relevant external, non-parametric knowledge as context for the generation process (Lewis et al., 2020). It helps mitigate key limitations such as hallucinations by grounding responses in more accurate, traceable, and up-to-date knowledge without parameter retraining (Borgeaud et al., 2022; Izacard et al., 2023; Karpukhin et al., 2020). Therefore, the effectiveness of RAG largely de-

pends on whether the retrieved knowledge is relevant, complete, and useful for answering the input query.

Standard RAG (Shao et al., 2023; Trivedi et al., 2023; Chen et al., 2024; Qiu et al., 2025) retrieves knowledge as text chunks based on vector similarity between the query and the chunks. However, each chunk is usually treated as an unstructured bag of tokens, which can obscure both intra- and inter-chunk semantic associations across related evidence. As a result, retrieval may miss relevant knowledge that is semantically connected either within a chunk or across chunks. To address this issue, recent approaches augment retrieval with semantic structures such as graphs. Graph-based RAG approaches (Sarathi et al., 2024; Edge et al., 2024; Cao et al., 2024; Hu et al., 2025; Zhu et al., 2025; Guo et al., 2025; Ma et al., 2025; Fang et al., 2025; Gutiérrez et al., 2024, 2025; Luo et al., 2026; Lien et al., 2026) use pre-built knowledge graphs (KGs) to explicitly capture associations between named entities in text chunks, and employ graph-based exploration methods, such as Personalized PageRank (Gutiérrez et al., 2024, 2025), to retrieve relevant knowledge. Despite their success in sense-making and multi-hop query answering, these methods are often query-agnostic: they retrieve all chunks associated with the mentioned entities and their nearby multi-hop neighbors regardless of what is asked in the query, introducing redundant information that can easily confuse the LLM. More recent work directly extracts knowledge from text chunks in the form of Question-Answer (QA) pairs (Yang et al., 2026). This enables more accurate matching between the user query and the relevant factual knowledge expressed in query formats. However, it still faces two limitations: (1) relying on text-based atomic facts extracted from chunks fails to preserve the inherent

<sup>†</sup>Corresponding author.

<sup>1</sup>The source code and data are available [here](#).

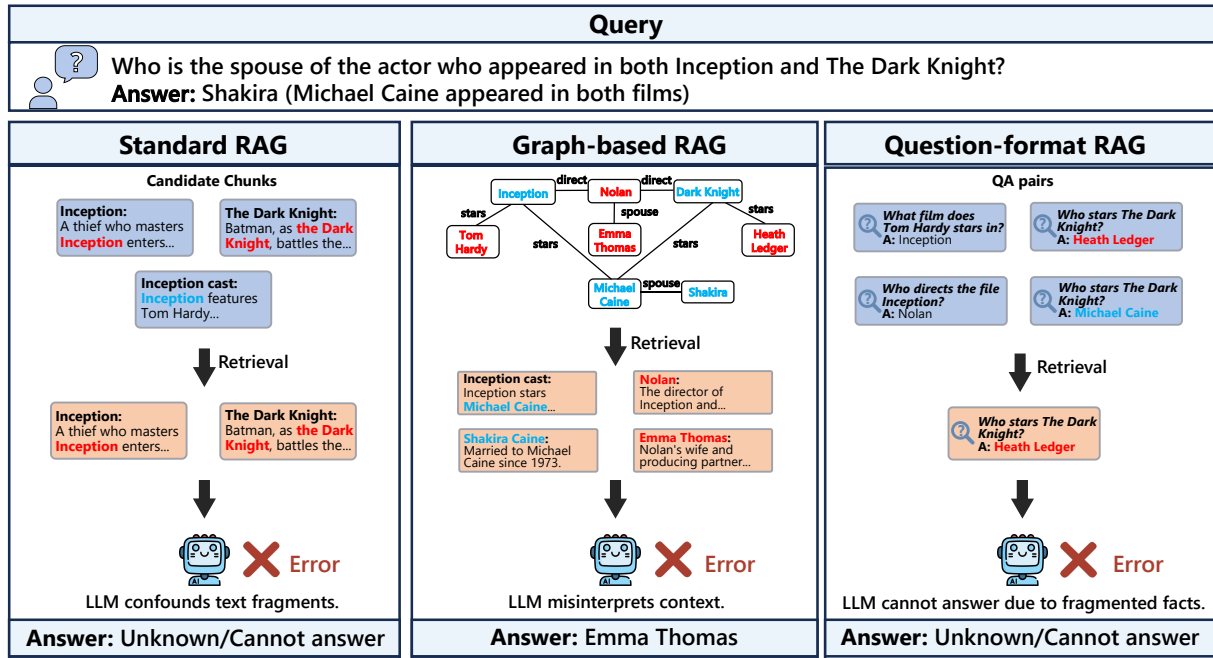


Figure 1: Retrieval limitations of current RAG methods, where red text denotes incorrectly retrieved knowledge for Standard RAG and knowledge that is not needed to directly answer the query for graph-based and question-format RAG, while blue text denotes knowledge that should be retrieved to answer the query. Standard RAG retrieves text chunks mentioning *Inception* and *The Dark Knight* through vector similarity, but disregards how the chunks are semantically related. Graph-based methods do not consider what the query specifically asks and may retrieve redundant information, such as the shared director of the two films, which can distract the reasoning process. Question-format methods improve query matching but may still miss or fragment key evidence: because QA pairs are directly converted from unstructured text chunk, the fact that Michael Caine starred in *Inception* may be missed; meanwhile, actor QA pairs for *The Dark Knight* remain fragmented, so retrieval may return incomplete actor evidence and miss the needed Michael Caine fact.

semantic associations among entities; (2) the generated QA pairs remain atomic and fragmented, making it difficult to reason over user queries that require multiple related atomic facts. We illustrate all these limitations in Figure 1.

To address these limitations, we propose **HolisticRAG**, which constructs a graph-grounded and query-aligned knowledge representation to improve RAG retrieval accuracy. Here, graph-grounded means that knowledge is anchored in graph structures that preserve semantic associations, while query-aligned means that it is represented in a question-based form consistent with user queries. Together, these properties enable HolisticRAG to compose complex question-format knowledge by aggregating multiple binary facts across chunks. Specifically, in the offline stage, HolisticRAG constructs question-based knowledge that preserves semantic associations through the KG while supporting more accurate retrieval through question-format representations. In the online stage, HolisticRAG adaptively performs

holistic knowledge clustering over the constructed questions to mitigate knowledge fragmentation and gather relevant evidence for answering the user query. Experiments across five open-domain and five closed-domain QA benchmarks demonstrate the effectiveness of HolisticRAG. Compared with the strongest baselines, HolisticRAG achieves relative improvements of up to **6.57%** in retrieval performance and up to **21.49%** in Exact Match and **13.93%** in F1 for downstream QA.

In summary, the contributions of this work are as follows:

- We develop a question-based knowledge representation that can preserve semantic associations and improve retrieval accuracy by grounding questions in graph relations and aligning them with user queries.
- We further introduce holistic knowledge clustering that can mitigate knowledge fragmentation by consolidating related fragmented knowledge during online retrieval.

- Extensive experiments demonstrate the effectiveness of our proposed framework.

## 2 Related Work

This section reviews standard and structure-augmented RAG, together with closely related retrieval methods that use generation for query–corpus alignment.

### 2.1 Standard RAG

Standard RAG retrieves knowledge as text chunks based on vector similarity between the user query and candidate chunks, and then provides the retrieved chunks to the LLM as external context. ITER-RETGEN (Shao et al., 2023) iteratively interleaves retrieval and generation, using the model’s output to guide further retrieval. IRCot (Trivedi et al., 2023) interleaves retrieval with chain-of-thought reasoning steps to improve multi-step QA. Propositionizer (Chen et al., 2024) indexes the corpus with atomic propositions instead of passages for finer-grained retrieval. Another method (Qiu et al., 2025) prioritizes low-entropy distributions from retrieved documents to mitigate distractibility. Despite their improvements, these methods still treat knowledge as unstructured or unit-level text, lacking explicit semantic associations across chunks.

### 2.2 Structure-augmented RAG

Structure-augmented RAG enriches retrieval with explicit intermediate structures, such as graphs and question-format memories. Graph-based methods typically construct text-derived knowledge graphs or graph-like structures to support knowledge reasoning. RAPTOR (Sarthi et al., 2024), GraphRAG (Edge et al., 2024), KG<sup>2</sup>RAG (Zhu et al., 2025), and LightRAG (Guo et al., 2025) organize corpus knowledge into hierarchical, summarized graph structures for retrieval and reasoning. ToG 2.0 (Ma et al., 2025) and KiRAG (Fang et al., 2025) further employ LLMs to perform reasoning over graph structures, while HippoRAG (Gutiérrez et al., 2024, 2025) uses a non-parametric Personalized PageRank algorithm for graph-based inference. HyperGraphRAG (Luo et al., 2026) and HyperRAG (Lien et al., 2026) further model higher-order relations over multiple entities or facts and use LLMs for reasoning. These methods better expose semantic associations than standard chunk retrieval, but their retrieval is often query-agnostic and may introduce redundant or weakly relevant evidence.

Inspired by earlier works on document-query alignment (Nogueira et al., 2019b,a; Gao et al., 2023), question-format methods improve query alignment by converting text into retrievable QA-style memories. CompactRAG (Yang et al., 2026), for example, decomposes text into atomic QA pairs for sub-query reasoning. However, these QA pairs remain isolated and may fragment complete knowledge.

## 3 Preliminaries

**RAG definition.** Given a user query  $q$  and a corpus  $\mathcal{C} = \{c_i\}_{i=1}^N$ , where each  $c_i$  denotes a text chunk, standard RAG first retrieves an evidence set  $\mathcal{E}_{\mathcal{C}}(q) \subseteq \mathcal{C}$  and then estimates the answer probability conditioned on this evidence:

$$P(a \mid q, \mathcal{C}) \approx P(a \mid q, \mathcal{E}_{\mathcal{C}}(q)) \quad (1)$$

**HolisticRAG definition.** HolisticRAG reformulates the evidence space as question-format memories. Given the corpus  $\mathcal{C}$ , HolisticRAG extracts triples  $\mathcal{T} = \{(s, r, o)\}$  and converts them into memory items  $\mathcal{M} = \{(q', t, c(t)) : q' \in \mathcal{Q}(t), t \in \mathcal{T}, c(t) \in \mathcal{C}\}$ , where  $\mathcal{Q}(t)$  denotes the questions generated from triple  $t$ ,  $q'$  is a generated question, and  $c(t)$  is the source chunk of  $t$ . HolisticRAG retrieves evidence  $\mathcal{E}_{\mathcal{M}}(q) \subseteq \mathcal{M}$  from this memory space and estimates the answer probability as:

$$P(a \mid q, \mathcal{M}) \approx P(a \mid q, \mathcal{E}_{\mathcal{M}}(q)) \quad (2)$$

## 4 HolisticRAG

As shown in Figure 2, HolisticRAG consists of four main components.

### 4.1 Fine-grained Fact Extraction

To establish semantic associations within each chunk, HolisticRAG represents entities and their factual relations as a graph rather than treating facts as isolated textual statements.

Existing RAG methods (Fang et al., 2025; Gutiérrez et al., 2024, 2025; Lien et al., 2026; Luo et al., 2026) often use a coarse two-stage pipeline: entity extraction followed by triple completion. Because relation discovery is folded into the final step, latent relations among extracted entities may be missed. To reduce this risk, we use a four-stage fine-grained LLM-based procedure for each chunk  $c_i$ : extract entities  $\mathcal{V}_i$ , identify candidate relations  $\mathcal{R}_i$ , match each relation  $r_i \in \mathcal{R}_i$  with plausible subjects to form  $\mathcal{P}_i = \{(s_i, r_i)\}$ , and complete graph-grounded triples  $\mathcal{T}_i = \{(s_i, r_i, o_i)\}$  by grounding

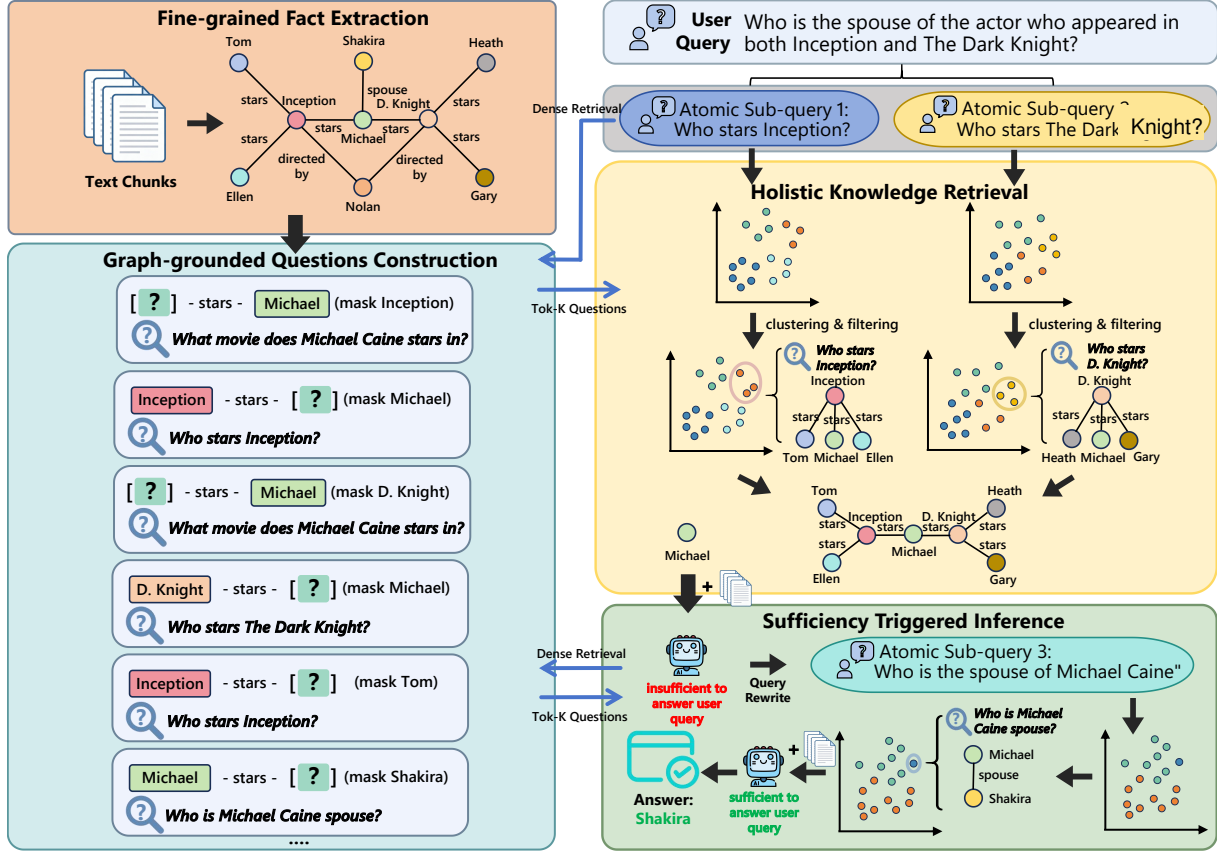


Figure 2: Overview of the proposed HolisticRAG framework. In the offline stage, HolisticRAG extracts fine-grained facts from text chunks and constructs graph-grounded questions to preserve semantic associations while enabling query-aligned retrieval. In the online stage, it performs holistic knowledge retrieval and sufficiency-triggered inference to consolidate fragmented evidence and gather sufficient supporting facts for answering the query.

$o_i$  in the original chunk. This decomposition makes relation discovery explicit and yields more robust graph-grounded triples.

## 4.2 Graph-Grounded Questions Construction

After constructing KG, HolisticRAG converts them into atomic questions. This step bridges the offline graph representation and online retrieval: KG triples preserve semantic associations within each chunk, while question-format memories enable query-aligned matching during online inference.

For each triple  $t_i = (s_i, r_i, o_i) \in \mathcal{T}_i$ , we use an LLM-based bilateral question generator to construct two atomic natural-language questions grounded in the source chunk  $c_i$ . Specifically, HolisticRAG generates a subject-masked question  $q_i^{(s)}$  that asks for  $s_i$  given  $(r_i, o_i)$ , and an object-masked question  $q_i^{(o)}$  that asks for  $o_i$  given  $(s_i, r_i)$ . The generated question set for chunk  $c_i$  is therefore  $\mathcal{Q}_i = \{q_i^{(s)}, q_i^{(o)} : t_i \in \mathcal{T}_i\}$ . Since each question is grounded in its corresponding triple, it still describes a single atomic fact. This bilateral gener-

ation allows each triple to be retrieved from either direction and helps preserve the completeness of chunk-level facts.

Each generated question  $q' \in \mathcal{Q}_i$  is linked back to its corresponding triple  $t_i$  and source chunk  $c_i$ , forming a memory item  $(q', t_i, c_i)$ . All memory items are then embedded into dense memories  $\mathcal{M}$  for retrieval.

## 4.3 Holistic Knowledge Retrieval

To address knowledge fragmentation, HolisticRAG performs clustering-based consolidation in the online stage.

Existing RAG methods (Lien et al., 2026; Luo et al., 2026) mitigate fragmentation by constructing chunk-level hyperedges offline, but this typically relies on LLMs and introduces substantial computational and token costs at corpus scale.

Online retrieval proceeds iteratively. Given the original query  $q$ , HolisticRAG first uses an LLM to identify one or more atomic sub-queries that

capture the initial information needs for solving  $q$ :

$$\mathcal{S}_1 = f_{\text{seed}}(q) = \{q_1^{(m)}\}_{m=1}^{M_1}, \quad (3)$$

where  $\mathcal{S}_1$  is the initial sub-query set and  $M_1$  is its size. At step  $k$ , let  $q_k$  be the current atomic sub-query. We embed  $q_k$  and retrieve the top- $K$  most similar generated questions from  $\mathcal{M}$ , with  $K$  discussed in Appendix B.3:

$$\mathcal{N}_k = \text{TopK}_K(q_k, \mathcal{M}; \text{sim}(q_k, q')) \quad (4)$$

where  $\mathcal{N}_k = \{(q'_j, t_j, c_j)\}_{j=1}^K$  contains retrieved questions  $\mathcal{Q}_k^{\text{ret}} = \{q'_j\}_{j=1}^K$ , linked triples  $\mathcal{T}_k^{\text{ret}} = \{t_j\}_{j=1}^K$ , and source chunks  $\mathcal{C}_k^{\text{ret}} = \{c_j\}_{j=1}^K$ . These questions are candidate fragmented knowledge pieces grounded in triples and chunks. Since related fragments may appear in both intra- and inter-chunk contexts, we apply Hierarchical Density-Based Spatial Clustering (McInnes et al., 2017) to their embeddings:

$$\Pi_k = \text{HDBSCAN}(\{\mathbf{e}(q') : q' \in \mathcal{Q}_k^{\text{ret}}\}) \quad (5)$$

where  $\Pi_k$  denotes the resulting clusters. Let  $q'_1$  be the top-1 retrieved question in  $\mathcal{N}_k$ , and let  $C_k(q'_1) \in \Pi_k$  be its cluster. We treat this cluster as relevant to  $q_k$  and use an LLM-based filter to remove irrelevant questions:

$$\mathcal{R}_k = f_{\text{filter}}(q_k, C_k(q'_1)) \quad (6)$$

If  $q'_1$  does not belong to any cluster, we set  $\mathcal{R}_k = \{q'_1\}$ . The remaining questions define the evidence for step  $k$ :

$$\mathcal{E}_k = \{(q', t, c) : (q', t, c) \in \mathcal{N}_k, q' \in \mathcal{R}_k\} \quad (7)$$

which contains the filtered questions, corresponding triples, and source chunks. If multiple initial sub-queries are generated, HolisticRAG applies the same retrieval and filtering process to each  $q_1^{(m)} \in \mathcal{S}_1$  and merges their evidence before LLM reasoning:

$$\mathcal{E}_1 = \bigcup_{m=1}^{M_1} \mathcal{E}_1^{(m)}, \quad (8)$$

where  $\mathcal{E}_1^{(m)}$  is the evidence for initial sub-query  $q_1^{(m)}$ . Thus, HolisticRAG consolidates fragmented knowledge required by the initial sub-query set while grounding evidence in both the graph structure and the original text.

#### 4.4 Sufficiency-Triggered Inference

Given accumulated evidence  $\mathcal{E}_{\leq k}$ , HolisticRAG asks an LLM whether it is sufficient to answer the original query  $q$ . If sufficient, the LLM outputs the final answer  $a$ ; otherwise, HolisticRAG generates a new atomic sub-query  $q_{k+1}$  conditioned on  $q$  and  $\mathcal{E}_{\leq k}$ :

$$q_{k+1} = f_{\text{rewrite}}(q, \mathcal{E}_{\leq k}) \quad (9)$$

The system then repeats retrieval and validation. Fallback is triggered in two cases: (1) two consecutive atomic sub-queries are nearly identical, i.e.,  $\cos(\mathbf{e}(q_k), \mathbf{e}(q_{k+1})) > \tau$ , indicating that rewriting fails to make progress; and (2) HolisticRAG cannot answer after  $D_{\text{max}}$  iterations. The choices of  $\tau$  and  $D_{\text{max}}$  are discussed in Appendix B.5 and Appendix B.4, respectively. In either case, HolisticRAG falls back to standard RAG: it embeds the original query  $q$ , retrieves the top-5 most similar text chunks from the full corpus, and asks the LLM to answer from them.

### 5 Experimental Setup

This section outlines the benchmarks, baselines, evaluation metrics, and implementation details used in our experiments.

#### 5.1 Benchmarks

We evaluate HolisticRAG on both open-domain and closed-domain QA benchmarks. For open-domain evaluation, we use two single-hop datasets, **NaturalQuestions (NQ)** (Kwiatkowski et al., 2019) and **PopQA** (Mallen et al., 2023), and three multi-hop datasets, **HotpotQA** (Yang et al., 2018), **2WikiMultihopQA** (Ho et al., 2020), and **MuSiQue** (Trivedi et al., 2022). Following HippoRAG 2.0 (Gutiérrez et al., 2025), we evaluate on sampled subsets of 1,000 queries per benchmark due to computational cost. For closed-domain evaluation, following HyperGraphRAG (Luo et al., 2026), we use five specialized benchmarks: **Agriculture**, **Computer Science (CS)**, **Legal**, **Medicine**, and **Mix**.

#### 5.2 Baselines

We compare HolisticRAG with representative baselines. For **Standard RAG**, we include **LLM2Vec-Llama-3-8B** (BehnamGhader et al., 2024), **E5-mistral-7b-instruct** (Wang et al., 2024), and **NV-Embed-v2 (7B)** (Lee et al., 2025) as dense retrievers. For **Structure-augmented RAG**, we include **RAPTOR** (Sarathi et al., 2024), **GraphRAG** (Edge

et al., 2024), **LightRAG** (Guo et al., 2025), **ToG 2.0** (Ma et al., 2025), **KiRAG** (Fang et al., 2025), **HippoRAG** (Gutiérrez et al., 2024), **HippoRAG 2.0** (Gutiérrez et al., 2025), **HyperGraphRAG** (Luo et al., 2026), **HyperRAG** (Lien et al., 2026) and **CompactRAG** (Yang et al., 2026).

### 5.3 Evaluation Metrics

To enable a consistent comparison across methods with different native retrieval granularities, we evaluate all baselines under the same setting and report Recall@1 on the open-domain benchmarks. For QA evaluation, we report Exact Match (EM) and F1 on all benchmarks, including closed-domain benchmarks where answer accuracy is the primary focus.

### 5.4 Implementation Details

We evaluate HolisticRAG with three LLM backbones, **Qwen2.5-32B-Instruct** (Yang et al., 2024), **Llama-3.3-70B-Instruct** (Grattafiori et al., 2025), and **GPT-4o** (Achiam et al., 2023), and three retrievers, **LLM2Vec-Llama-3-8B** (BehnamGhader et al., 2024), **E5-mistral-7b-instruct** (Wang et al., 2024), and **NV-Embed-v2 (7B)** (Lee et al., 2025). Qwen2.5-32B-Instruct and Llama-3.3-70B-Instruct are served with vLLM, while GPT-4o is accessed through the Azure OpenAI API. All experiments run on eight NVIDIA H200 140GB GPUs, and prompts are provided in Appendix D. For fair comparison, baselines use the same LLM backbones when applicable and follow the default hyperparameters reported in their original papers.

## 6 Experimental Results

This section presents the main experimental results and analysis. Unless otherwise specified, all experiments use **Qwen2.5-32B-Instruct** as the LLM backbone and **NV-Embed-v2 (7B)** as the retriever, and all baselines are evaluated under the same setting when applicable. We first compare HolisticRAG with existing methods on open-domain and closed-domain QA, and then analyze the contribution of each component through ablation studies. We further evaluate the effectiveness of the holistic knowledge retrieval mechanism, provide a case analysis, and analyze the robustness of HolisticRAG under corpus expansion.

### 6.1 Main Results

On open-domain benchmarks (Tables 1 and 2), HolisticRAG achieves the best Recall@1 on all five

datasets under the unified evaluation setting, outperforming the strongest baseline by 0.02–4.14 points. For downstream QA, HolisticRAG achieves the best result on 9/10 EM/F1 metrics. The gains are especially clear on multi-hop datasets, where HolisticRAG obtains an F1 of 72.63 on HotpotQA, 59.36 on 2WikiMultihopQA, and 51.68 on MuSiQue. This suggests that graph-grounded question-format memories and adaptive evidence consolidation are particularly useful for questions requiring multiple related facts. The only exception is PopQA F1, where CompactRAG performs slightly better, likely because many PopQA questions are short single-hop entity queries.

On closed-domain benchmarks (Table 3), HolisticRAG outperforms HyperGraphRAG on 8/10 EM/F1 metrics, showing strong transfer to domain-specific QA. HyperGraphRAG performs better on Mix, likely because this heterogeneous benchmark benefits from broader hypergraph expansion, whereas HolisticRAG uses more conservative adaptive filtering to avoid redundant evidence. Results with larger LLM backbones show consistent trends, as reported in Appendix B.1.

### 6.2 Ablation Study

We conduct a set of ablation study on HotpotQA to examine both the knowledge representation and the online retrieval design in HolisticRAG. To isolate the contribution of the question-format representation from the iterative retrieval mechanism, we keep the seed generation, retrieval, clustering, sufficiency judgment, and query-rewriting loop fixed, while replacing the question-format memory with either raw text chunks (Raw Chunks) or linearized KG triples (KG Triples). We also remove fine-grained fact extraction (FFE) or bilateral question construction (BQC), replace knowledge clustering with direct top-1, top-3, or top-5 selection, and remove the post-clustering filter (CF). Retrieval is reported using Recall@1. As shown in Table 4, the full model achieves the best performance, with 48.86 Recall@1 and 72.63 F1. Replacing question-format memories with raw chunks or KG triples reduces F1 to 67.41 and 68.92, respectively, despite using the same online loop. The remaining ablations also underperform the full model. These results show that the gains arise from both the graph-grounded, query-aligned representation and the iterative evidence-consolidation mechanism, rather than from the online loop alone.

| Method                                         | NQ                       | PopQA                    | HotpotQA                 | 2WikiMultihopQA          | MuSiQue                  |
|------------------------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|
| LLM2Vec-Llama-3-8B (BehnamGhader et al., 2024) | 64.29                    | 48.76                    | 40.41                    | 42.23                    | 40.24                    |
| E5-mistral-7b-instruct (Wang et al., 2024)     | 66.59                    | 50.87                    | 42.23                    | 41.25                    | 37.58                    |
| NV-Embed-v2 (7B) (Lee et al., 2025)            | 67.33                    | 51.03                    | 46.08                    | 40.37                    | 39.26                    |
| RAPTOR (Sarhi et al., 2024)                    | 57.83                    | 48.33                    | 42.38                    | 39.87                    | 28.75                    |
| GraphRAG (Edge et al., 2024)                   | 63.23                    | 49.87                    | 40.15                    | 39.64                    | 34.28                    |
| LightRAG (Guo et al., 2025)                    | 59.64                    | 46.31                    | 37.82                    | 36.15                    | 31.53                    |
| ToG 2.0 (Ma et al., 2025)                      | 65.87                    | 51.34                    | 42.67                    | 41.82                    | 37.24                    |
| KiRAG (Fang et al., 2025)                      | 65.35                    | 48.56                    | 43.03                    | 45.33                    | 38.72                    |
| HippoRAG (Gutiérrez et al., 2024)              | 51.61                    | <u>55.23</u>             | 41.87                    | 45.21                    | 40.33                    |
| HippoRAG 2.0 (Gutiérrez et al., 2025)          | <u>71.21</u>             | 53.31                    | <u>46.65</u>             | <u>47.20</u>             | <u>41.54</u>             |
| HyperGraphRAG (Luo et al., 2026)               | 66.78                    | 52.46                    | 43.59                    | 42.87                    | 38.42                    |
| HyperRAG (Lien et al., 2026)                   | 64.15                    | 50.72                    | 41.38                    | 40.96                    | 36.57                    |
| CompactRAG (Yang et al., 2026)                 | 67.42                    | 53.18                    | 44.76                    | 43.55                    | 39.13                    |
| <b>HolisticRAG (Ours)</b>                      | <b>75.35<sup>†</sup></b> | <b>56.05<sup>†</sup></b> | <b>48.86<sup>†</sup></b> | <b>47.22<sup>†</sup></b> | <b>44.27<sup>†</sup></b> |

Table 1: Open-domain retrieval performance measured by Recall@1. Bold and underline denote the best and second-best results in each column, respectively. † indicates HolisticRAG significantly outperforms the best baseline under a paired Wilcoxon signed-rank test with  $p < 0.05$ .

| Method                                         | NQ                       |                          | PopQA                    |              | HotpotQA                 |                          | 2WikiMultihopQA          |                          | MuSiQue                  |                          |
|------------------------------------------------|--------------------------|--------------------------|--------------------------|--------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|
|                                                | EM                       | F1                       | EM                       | F1           | EM                       | F1                       | EM                       | F1                       | EM                       | F1                       |
| LLM2Vec-Llama-3-8B (BehnamGhader et al., 2024) | 41.26                    | 57.39                    | 31.22                    | 51.29        | 41.28                    | 65.26                    | 25.16                    | 56.87                    | 27.78                    | 43.25                    |
| E5-mistral-7b-instruct (Wang et al., 2024)     | 41.24                    | 55.29                    | 29.83                    | 54.29        | 49.21                    | 68.29                    | 25.67                    | <u>57.45</u>             | 24.21                    | <u>45.36</u>             |
| NV-Embed-v2 (7B) (Lee et al., 2025)            | 40.40                    | 58.88                    | 30.75                    | 53.27        | 47.45                    | <u>70.37</u>             | 22.45                    | 57.32                    | 28.84                    | 42.39                    |
| RAPTOR (Sarhi et al., 2024)                    | 42.78                    | 47.38                    | 30.47                    | 53.38        | 46.59                    | 64.29                    | 34.62                    | 50.36                    | 29.97                    | 26.31                    |
| GraphRAG (Edge et al., 2024)                   | 39.45                    | 45.23                    | 25.67                    | 49.56        | 48.78                    | 63.45                    | 32.71                    | 54.67                    | 22.59                    | 35.50                    |
| LightRAG (Guo et al., 2025)                    | 3.34                     | 20.03                    | 1.24                     | 4.67         | 0.92                     | 3.29                     | 1.56                     | 8.68                     | 0.76                     | 2.47                     |
| ToG 2.0 (Ma et al., 2025)                      | 43.27                    | 49.98                    | 29.30                    | 52.13        | 51.33                    | 57.26                    | 31.49                    | 48.52                    | <u>30.29</u>             | 43.76                    |
| KiRAG (Fang et al., 2025)                      | 46.28                    | 54.01                    | 27.36                    | 50.37        | 46.28                    | 62.81                    | 32.33                    | 54.89                    | 23.34                    | 30.70                    |
| HippoRAG (Gutiérrez et al., 2024)              | 41.26                    | 51.29                    | 22.59                    | 50.28        | 52.37                    | 63.89                    | 38.17                    | 45.35                    | 23.88                    | 34.14                    |
| HippoRAG 2.0 (Gutiérrez et al., 2025)          | <u>46.89</u>             | <u>60.33</u>             | <u>31.47</u>             | 48.88        | <u>56.20</u>             | 68.86                    | <u>38.36</u>             | 47.57                    | 25.21                    | 38.64                    |
| HyperRAG (Lien et al., 2026)                   | 40.31                    | 48.96                    | 20.46                    | 49.21        | 51.38                    | 60.97                    | 35.86                    | 43.64                    | 22.71                    | 31.48                    |
| HyperGraphRAG (Luo et al., 2026)               | 42.08                    | 50.74                    | 21.91                    | 51.36        | 53.02                    | 63.41                    | 37.44                    | 46.02                    | 24.53                    | 33.62                    |
| CompactRAG (Yang et al., 2026)                 | 43.31                    | 56.78                    | 28.81                    | <b>56.06</b> | 48.50                    | 59.75                    | 23.80                    | 26.58                    | 22.70                    | 33.71                    |
| <b>HolisticRAG (Ours)</b>                      | <b>50.39<sup>†</sup></b> | <b>63.39<sup>†</sup></b> | <b>35.47<sup>†</sup></b> | <u>54.31</u> | <b>59.00<sup>†</sup></b> | <b>72.63<sup>†</sup></b> | <b>43.80<sup>†</sup></b> | <b>59.36<sup>†</sup></b> | <b>36.80<sup>†</sup></b> | <b>51.68<sup>†</sup></b> |

Table 2: Open-domain QA performance. For HyperRAG, we report the better result between its HyperRetriever and HyperMemory settings. Bold and underline denote the best and second-best results, respectively. † indicates HolisticRAG significantly outperforms the best baseline under a paired Wilcoxon signed-rank test with  $p < 0.05$ .

| Method                           | Agriculture              |                          | CS                       |                          | Legal                    |                          | Mix          |              | Medicine                 |                          |
|----------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------|--------------|--------------------------|--------------------------|
|                                  | EM                       | F1                       | EM                       | F1                       | EM                       | F1                       | EM           | F1           | EM                       | F1                       |
| HyperGraphRAG (Luo et al., 2026) | 15.87                    | 22.82                    | 9.80                     | 20.37                    | 17.89                    | 29.55                    | <b>23.64</b> | <b>32.51</b> | 15.30                    | 30.16                    |
| <b>HolisticRAG (Ours)</b>        | <b>16.32<sup>†</sup></b> | <b>23.45<sup>†</sup></b> | <b>15.32<sup>†</sup></b> | <b>26.81<sup>†</sup></b> | <b>23.56<sup>†</sup></b> | <b>35.31<sup>†</sup></b> | 17.88        | 29.89        | <b>16.39<sup>†</sup></b> | <b>32.75<sup>†</sup></b> |

Table 3: Closed-domain QA performance on five benchmarks, comparing HolisticRAG with HyperGraphRAG. Bold and underline denote the best and second-best results, respectively. † indicates HolisticRAG significantly outperforms the best baseline under a paired Wilcoxon signed-rank test with  $p < 0.05$ .

### 6.3 Effectiveness of Holistic Knowledge Retrieval

We compare HolisticRAG’s online clustering strategy with an offline alternative that pre-clusters all generated questions globally to demonstrate the effectiveness of our design choice. Since global clustering over all generated questions is expensive, we first apply PCA to reduce question em-

bedding dimensionality before clustering. As Table 5 shows, offline clustering is computationally expensive: 13–20 minutes on four benchmarks and over one hour on MuSiQue. While it performs better on the two single-hop benchmarks, it underperforms online clustering on all three multi-hop datasets, where fragmented evidence must be consolidated. This is likely because global clusters

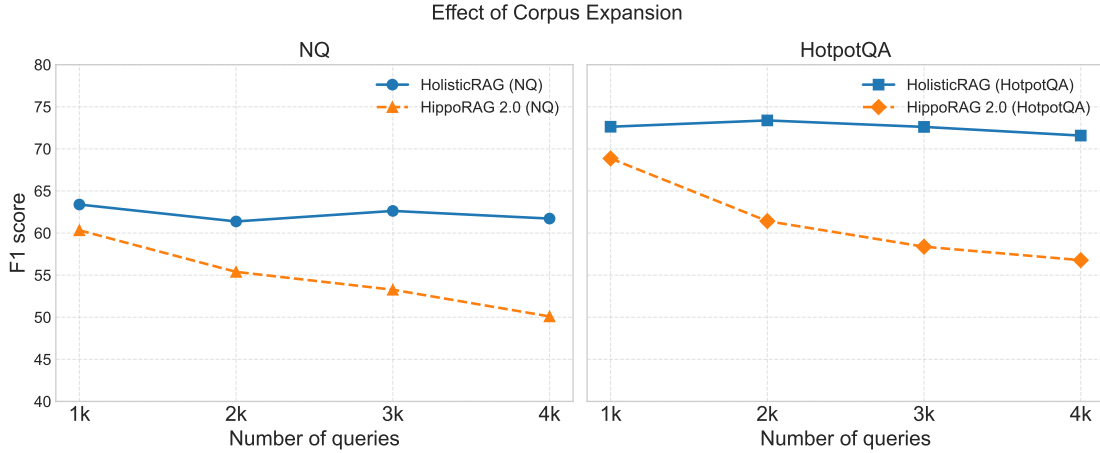


Figure 3: QA performance under corpus expansion on NQ and HotpotQA. HolisticRAG is compared with HippoRAG 2.0 as the retrieval corpus grows from 1,000 to 4,000 queries.

| Method                    | Recall@1     | F1           |
|---------------------------|--------------|--------------|
| Raw Chunks                | 43.24        | 67.41        |
| KG Triples                | 45.91        | 68.92        |
| w/o FFE                   | 42.36        | 65.37        |
| w/o BQC                   | 44.28        | 66.87        |
| w/o Clustering (top-1)    | 42.51        | 65.77        |
| w/o Clustering (top-3)    | 44.73        | 67.82        |
| w/o Clustering (top-5)    | 45.12        | 65.71        |
| w/o CF                    | 46.85        | 69.88        |
| <b>HolisticRAG (Full)</b> | <b>48.86</b> | <b>72.63</b> |

Table 4: Ablation study on HotpotQA. Bold values indicate the best performance in each column.

| Benchmark       | Off. Time       | On. Time | Off. F1 | On. F1 |
|-----------------|-----------------|----------|---------|--------|
| NQ              | 13 min 43 s     | 0.09 s   | 63.38   | 58.37  |
| PopQA           | 14 min 27 s     | 0.06 s   | 54.29   | 49.97  |
| HotpotQA        | 16 min 59 s     | 0.09 s   | 64.38   | 72.63  |
| 2WikiMultihopQA | 20 min 12 s     | 0.05 s   | 48.32   | 59.36  |
| MuSiQue         | 1 h 05 min 37 s | 0.12 s   | 45.97   | 51.68  |

Table 5: Comparison between offline global clustering and HolisticRAG’s online query-conditioned clustering across single-hop and multi-hop benchmarks.

are query-agnostic and can become broad and noisy. In contrast, HolisticRAG clusters only the retrieved candidates for each atomic query, adding at most 0.12 seconds per query while achieving much higher F1 on HotpotQA, 2WikiMultihopQA, and MuSiQue. These results support our choice of query-aware online consolidation conditioned on the current reasoning context. We further provide cost and efficiency analysis in Appendix B.2.

#### 6.4 Case Analysis

We use representative HotpotQA cases to illustrate HolisticRAG’s offline preservation of semantic associations and online mitigation of knowledge fragmentation. Unlike CompactRAG (Yang et al., 2026), which stores isolated QA pairs, HolisticRAG stores question-triple pairs that keep each retrievable question grounded in its corresponding graph relation. For the two selected cases, each QA pair produced by CompactRAG has a corresponding question-triple form in HolisticRAG, enabling a direct comparison of what additional relational information is preserved. Table 6 compares selected outputs from both methods. The cases show two key benefits. First, HolisticRAG retains graph-grounded details that CompactRAG misses, such as the recognizer “Muller-Doblies” for Hymenocallideae and type facts like “Hymenocallideae is a tribe” and “Clinantheae is a tribe.” Second, for set-oriented queries such as *What tribes are there under Amaryllidaceae?*, the answer requires combining facts from multiple chunks. HolisticRAG can cluster related question-triple pairs to gather both Hymenocallideae and Clinantheae as tribes under Amaryllidaceae, whereas isolated QA pairs make this consolidation harder.

#### 6.5 Effect of Corpus Expansion

We study how QA performance changes as the corpus becomes larger and noisier. Starting from the original 1,000-query setting on the NQ and HotpotQA development benchmarks, we additionally sample 3,000 non-overlapping queries from each benchmark (development set) and expand the corpus to 2,000, 3,000, and 4,000 queries. We

| Source Chunk                                                                                                                                                                                                                                         | CompactRAG QA Pairs                                                                                                   | HolisticRAG Question-Triple Pairs                                                                                                                             |
|------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|-----------------------------------------------------------------------------------------------------------------------|---------------------------------------------------------------------------------------------------------------------------------------------------------------|
| Hymenocallideae is a tribe (in the family Amaryllidaceae, subfamily Amaryllidoideae), where it forms part of the Andean clade, one of two American clades. The tribe was originally recognised by both Meerow (1995) and the Muller-Doblies' (1996). | Q: What tribe is there under Amaryllidaceae?<br>A: Hymenocallideae<br>Q: Who recognized Hymenocallideae?<br>A: Meerow | Q: What is Hymenocallideae?<br>T: Hymenocallideae - is a - tribe<br>Q: Who recognized Hymenocallideae?<br>T: Hymenocallideae - recognized by - Muller-Doblies |
| Clinantheae is a tribe (in the family Amaryllidaceae, subfamily Amaryllidoideae), where it forms part of the Andean clade, one of two American clades. The tribe was described in 2000 by Alan Meerow.                                               | Q: What tribe is there under Amaryllidaceae?<br>A: Clinantheae<br>Q: Who has described Clinantheae?<br>A: Alan Meerow | Q: What is Clinantheae?<br>T: Clinantheae - is a - tribe<br>Q: What tribe was described by Alan Meerow?<br>T: Clinantheae - described by - Alan Meerow        |

Table 6: Qualitative comparison of selected CompactRAG QA pairs and HolisticRAG question-triple pairs generated from two HotpotQA chunks.

compare HolisticRAG with the strongest baseline, HippoRAG 2.0. As shown in Figure 3, HolisticRAG consistently achieves higher F1 across all corpus sizes. More importantly, HolisticRAG maintains relatively stable performance as the corpus expands, whereas HippoRAG 2.0 shows a clear downward trend on both NQ and HotpotQA. These results indicate that HolisticRAG better preserves answer quality as additional queries introduce more distractor evidence.

## 7 Conclusion

In this paper, we introduce HolisticRAG, a framework designed to retrieve evidence that is both semantically connected and aligned with the user query. HolisticRAG builds graph-grounded question-format memories that retain factual relations while supporting query-oriented matching. It further performs online holistic knowledge clustering to consolidate fragmented evidence and adaptively collect the supporting facts needed for answering complex queries. Experiments on open-domain and closed-domain QA benchmarks show that this design improves both retrieval and downstream answer accuracy.

## Limitations

Although HolisticRAG improves retrieval and QA performance, it still has several limitations. First, its offline knowledge construction relies on LLM-based fact extraction and question generation, so extraction errors or incomplete question-triple pairs may affect downstream retrieval. Second, online clustering improves evidence consolidation but introduces additional inference costs associated with clustering and filtering. Finally, our evaluation cov-

ers representative open-domain and closed-domain QA benchmarks, but further studies on multilingual, rapidly changing, and highly specialized corpora are needed.

## References

- Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altschmidt, Sam Altman, Shyamal Anadkat, and 1 others. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.
- Parishad BehnamGhader, Vaibhav Adlakha, Marius Mosbach, Dzmitry Bahdanau, Nicolas Chapados, and Siva Reddy. 2024. Llm2vec: Large language models are secretly powerful text encoders. *arXiv preprint arXiv:2404.05961*.
- Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George Bm Van Den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, and 1 others. 2022. Improving language models by retrieving from trillions of tokens. In *International conference on machine learning*, pages 2206–2240. PMLR.
- Yukun Cao, Zengyi Gao, Zhiyang Li, Xike Xie, S Kevin Zhou, and Jianliang Xu. 2024. Lego-graphrag: Modularizing graph-based retrieval-augmented generation for design space exploration. *arXiv preprint arXiv:2411.05844*.
- Tong Chen, Hongwei Wang, Sihao Chen, Wenhao Yu, Kaixin Ma, Xinran Zhao, Hongming Zhang, and Dong Yu. 2024. Dense x retrieval: What retrieval granularity should we use? In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pages 15159–15177.
- 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*.
- Jinyuan Fang, Zaiqiao Meng, and Craig Macdonald. 2025. Kirag: Knowledge-driven iterative retriever for enhancing retrieval-augmented generation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 18969–18985.
- Luyu Gao, Xueguang Ma, Jimmy Lin, and Jamie Callan. 2023. Precise zero-shot dense retrieval without relevance labels. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 1762–1777.
- Aaron Grattafiori, Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Alex Vaughan, and 1 others. 2025. The llama 3 herd of models.
- Zirui Guo, Lianghao Xia, Yanhua Yu, Tian Ao, and Chao Huang. 2025. Lightrag: Simple and fast retrieval-augmented generation. In *EMNLP (Findings)*, pages 10746–10761.
- Bernal J Gutiérrez, Yiheng Shu, Yu Gu, Michihiro Yasunaga, and Yu Su. 2024. Hipporag: Neurobiologically inspired long-term memory for large language models. *Advances in neural information processing systems*, 37:59532–59569.
- 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. *arXiv preprint arXiv:2502.14802*.
- Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. 2020. Constructing a multi-hop qa dataset for comprehensive evaluation of reasoning steps. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6609–6625.
- Yuntong Hu, Zhihan Lei, Zheng Zhang, Bo Pan, Chen Ling, and Liang Zhao. 2025. Grag: Graph retrieval-augmented generation. In *Findings of the association for computational linguistics: NAACL 2025*, pages 4145–4157.
- Gautier Izacard, Patrick Lewis, Maria Lomeli, Lucas Hosseini, Fabio Petroni, Timo Schick, Jane Dwivedi-Yu, Armand Joulin, Sebastian Riedel, and Edouard Grave. 2023. Atlas: Few-shot learning with retrieval augmented language models. *Journal of Machine Learning Research*, 24(251):1–43.
- Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 conference on empirical methods in natural language processing (EMNLP)*, pages 6769–6781.
- Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, and 1 others. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:453–466.
- Chankyu Lee, Rajarshi Roy, Mengyao Xu, Jonathan Raiman, Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. 2025. Nv-embed: Improved techniques for training llms as generalist embedding models. In *International Conference on Learning Representations*, volume 2025, pages 79310–79333.
- Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, and 1 others. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33:9459–9474.
- Wen-Sheng Lien, Yu-Kai Chan, Hao-Lung Hsiao, Bo-Kai Ruan, Meng-Fen Chiang, Chien-An Chen, Yi-Ren Yeh, and Hong-Han Shuai. 2026. Hyperrag: Reasoning n-ary facts over hypergraphs for retrieval augmented generation. In *Proceedings of the ACM Web Conference 2026*, pages 2465–2476.
- Haoran Luo, Guanting Chen, Yandan Zheng, Xiaobao Wu, Yikai Guo, Qika Lin, Yu Feng, Zemin Kuang, Meina Song, Yifan Zhu, and 1 others. 2026. Hypergraphrag: Retrieval-augmented generation via hypergraph-structured knowledge representation. *Advances in Neural Information Processing Systems*, 38:152206–152234.
- 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*, volume 2025, pages 52782–52806.
- Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. 2023. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In *Proceedings of the 61st annual meeting of the association for computational linguistics (volume 1: Long papers)*, pages 9802–9822.
- Leland McInnes, John Healy, Steve Astels, and 1 others. 2017. hdbscan: Hierarchical density based clustering. *J. Open Source Softw.*, 2(11):205.
- Rodrigo Nogueira, Jimmy Lin, and AI Epistemic. 2019a. From doc2query to docttttquery. *Online preprint*, 6(2):1–3.
- Rodrigo Nogueira, Wei Yang, Jimmy Lin, and Kyunghyun Cho. 2019b. Document expansion by query prediction. *arXiv preprint arXiv:1904.08375*.

- Zexuan Qiu, Zijing Ou, Bin Wu, Jingjing Li, Aiwei Liu, and Irwin King. 2025. Entropy-based decoding for retrieval-augmented large language models. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 4616–4627.
- Parth Sarthi, Salman Abdullah, Aditi Tuli, Shubh Khanna, Anna Goldie, and Christopher Manning. 2024. Raptor: Recursive abstractive processing for tree-organized retrieval. In *International Conference on Learning Representations*, volume 2024, pages 32628–32649.
- Zhihong Shao, Yeyun Gong, Yelong Shen, Minlie Huang, Nan Duan, and Weizhu Chen. 2023. Enhancing retrieval-augmented large language models with iterative retrieval-generation synergy. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 9248–9274.
- Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2022. musique: Multi-hop questions via single-hop question composition. *Transactions of the Association for Computational Linguistics*, 10:539–554.
- Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2023. Interleaving retrieval with chain-of-thought reasoning for knowledge-intensive multi-step questions. In *Proceedings of the 61st annual meeting of the association for computational linguistics (volume 1: long papers)*, pages 10014–10037.
- Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. 2024. Improving text embeddings with large language models. In *Proceedings of the 62nd annual meeting of the association for computational linguistics (volume 1: long papers)*, pages 11897–11916.
- Hao Yang, Zhiyu Yang, Xupeng Zhang, Wei Wei, Yunjie Zhang, and Lin Yang. 2026. Compactrag: Reducing llm calls and token overhead in multi-hop question answering. In *Proceedings of the ACM Web Conference 2026*, pages 2240–2251.
- Qwen An Yang, Baosong Yang, Beichen Zhang, Binyuan Hui, Bo Zheng, Bowen Yu, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Yang, Jiaxin Yang, Jingren Zhou, Junyang Lin, and 25 others. 2024. [Qwen2.5 technical report](#). *ArXiv*, abs/2412.15115.
- 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 2018 conference on empirical methods in natural language processing*, pages 2369–2380.
- Xiangrong Zhu, Yuexiang Xie, Yi Liu, Yaliang Li, and Wei Hu. 2025. Knowledge graph-guided retrieval augmented generation. In *Proceedings of the 2025 Conference of the Nations of the Americas Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 8912–8924.

## A Detailed Data Statistics

We report the statistics of the generated questions used by HolisticRAG on both open-domain and closed-domain benchmarks. Tables 7 and 8 show the total number of generated questions and the average number of tokens per generated question. The open-domain multi-hop benchmarks generally contain more generated questions and longer generated questions than single-hop benchmarks, with MuSiQue producing the largest question set. The closed-domain benchmarks are smaller in scale, but still show domain-dependent variation in question length.

## B Additional Experimental Results

### B.1 Performance under Alternative Model Configurations

Tables 9, 10, and 11 report the results using **Llama-3.3-70B-Instruct** as the LLM backbone and **NV-Embed-v2 (7B)** as the retriever. The results indicate that HolisticRAG remains best in Recall@1 on all five datasets. For downstream QA, HolisticRAG obtains the best result on eight out of ten EM/F1 metrics. The two exceptions are PopQA F1 and HotpotQA F1: CompactRAG remains strong on short single-hop entity queries, while HippoRAG 2.0 benefits from entity-centric expansion on bridge-style HotpotQA questions. On closed-domain benchmarks, HolisticRAG outperforms HyperGraphRAG on seven out of ten EM/F1 metrics, including both metrics on Agriculture, CS, and Mix, while HyperGraphRAG remains stronger on Legal EM/F1 and Medicine EM.

Tables 12, 13, and 14 report the results using **GPT-4o** as the LLM backbone and **NV-Embed-v2 (7B)** as the retriever. The results likewise indicate that HolisticRAG remains best in Recall@1 on all five open-domain datasets. In open-domain QA, HolisticRAG achieves the best result on all ten EM/F1 metrics, indicating that the retrieved evidence is highly useful for answer generation with GPT-4o. On closed-domain benchmarks, HolisticRAG and HyperGraphRAG each achieve the best score on five out of ten metrics: HolisticRAG is stronger on all F1 metrics except Mix, while HyperGraphRAG remains competitive on EM, especially for CS, Legal, Mix, and Medicine. This suggests that HolisticRAG improves semantic answer quality, whereas exact-match performance in specialized domains can still benefit from broader hypergraph expansion.

### B.2 Cost and Efficiency

We further compare the cost and efficiency of HolisticRAG with representative baselines on HotpotQA. All methods use **NV-Embed-v2 (7B)** as the retriever, and we evaluate three LLM backbones: **Qwen2.5-32B-Instruct**, **Llama-3.3-70B-Instruct**, and **GPT-4o**. We report four metrics: average offline processing time per chunk, average offline token consumption per chunk, average online processing time per query, and average online token consumption per query. Tables 15, 16, and 17 show the results. Overall, HolisticRAG is not the most efficient method, but it is also far from the most expensive one. Its offline cost is moderate among graph- and query-augmented methods, and its online latency remains within the middle range despite performing adaptive evidence consolidation. This indicates that HolisticRAG improves retrieval and QA effectiveness without introducing prohibitive computational overhead.

### B.3 Effect of the Candidate Question Number

We further validate the number of retrieved candidate questions before graph-grounded clustering. In the main setting, HolisticRAG retrieves the top-20 generated questions for each atomic query and then applies clustering to consolidate related question-format knowledge. To examine this choice, we vary the candidate number among 10, 15, 20, 25, and 30, and evaluate HolisticRAG on HotpotQA using **Qwen2.5-32B-Instruct** as the LLM backbone and **NV-Embed-v2 (7B)** as the retriever. Table 18 shows that retrieving 20 candidate questions achieves the best F1 score. Smaller values may miss relevant question-format knowledge before clustering, while larger values may introduce more noisy candidates and reduce clustering precision. Therefore, we use top-20 candidate questions in the main experiments.

### B.4 Effect of the Maximum Inference Depth

We also investigate why HolisticRAG uses a maximum online inference depth of 5. Specifically, we vary the maximum depth from 1 to 8 and evaluate QA F1 on HotpotQA using **Qwen2.5-32B-Instruct** as the LLM backbone and **NV-Embed-v2 (7B)** as the retriever. Table 19 shows that performance improves substantially when the depth increases from 1 to 5, suggesting that shallow reasoning may stop before collecting sufficient multi-hop evidence. The best performance is achieved at depth 5. Fur-

ther increasing the depth degrades F1, likely because additional iterations introduce redundant or noisy evidence and increase the chance of query drift. Therefore, we set the maximum online inference depth to 5 in the main experiments.

## B.5 Effect of the Fallback Threshold

We also study the choice of the fallback threshold used to detect nearly repeated atomic queries. Specifically, we vary the threshold among 0.95, 0.90, 0.85, and 0.80, and evaluate HolisticRAG on HotpotQA using **Qwen2.5-32B-Instruct** as the LLM backbone and **NV-Embed-v2 (7B)** as the retriever. Table 20 reports the F1 scores under different threshold settings. The best performance is achieved when the threshold is set to 0.90. Compared with 0.95, lowering the threshold to 0.90 slightly improves F1, suggesting that a moderately stricter fallback condition can better avoid ineffective repeated retrieval steps. However, further lowering the threshold to 0.85 or 0.80 leads to a larger performance drop, indicating that overly aggressive fallback may interrupt useful multi-step reasoning. Therefore, we set the fallback threshold to 0.90 in our main experiments.

## C Demonstrations

### C.1 Seed Query Generation Examples

Table 21 presents ten examples sampled from HotpotQA, where Qwen2.5-32B-Instruct is used to generate atomic seed queries from the original complex queries.

### C.2 Reasoning Example

The following example shows how HolisticRAG answers a HotpotQA query using **Qwen2.5-32B-Instruct** as the LLM backbone and **NV-Embed-v2 (7B)** as the retriever. Starting from the original complex query, the model first generates an atomic seed query, retrieves the corresponding evidence, and then produces the final answer based on the retrieved passage and triple-level evidence.

#### HolisticRAG reasoning example

**Original query:** Works by Hanna Leena Kristiina Varis are part of a collection in a museum that houses approximately 65,000 what?

**Seed query:** What type of items does the museum that houses works by Hanna Leena Kristiina Varis collect?

**Round 1 generated query:** What is the approximate number of items in the collection of the Wäinö

Aaltonen Museum of Art?

**Retrieved passage:** The Albertina is a museum in the Innere Stadt (First District) of Vienna, Austria. It houses one of the largest and most important print rooms in the world with approximately 65,000 drawings and approximately 1 million old master prints, as well as more modern graphic works, photographs and architectural drawings. Apart from the graphics collection the museum has recently acquired on permanent loan two significant collections of Impressionist and early 20th-century art, some of which will be on permanent display. The museum also houses temporary exhibitions.

**Retrieved triple:** (Albertina, houses, print rooms, drawings, old master prints, modern graphic works, photographs, architectural drawings)

**Final answer:** drawings

## D Prompt Templates

### D.1 Triple Fact Extraction

#### D.1.1 Entity Extraction

##### Entity Extraction

**Goal:** Extract named entities from a given paragraph.

##### System Prompt

# Role  
You are an expert Information Extraction system specialized in Named Entity Recognition (NER).

# Tasks  
Identify and extract all unique named entities from the provided text. This includes, but is not limited to:  
- Persons (Actors, Characters, Directors)  
- Organizations (Studios, Radio Stations)  
- Locations (Countries, Cities)  
- Dates/Time periods  
- Creative Works (Movies, Songs, Portals)

# Constraints  
- Output ONLY a valid JSON object.  
- Key: "named\_entities", Value: A list of strings.  
- Each entity must be unique.  
- Do not include any explanation or conversational text.

##### Task

Given a paragraph, output a JSON object containing the list of named entities.

##### Example

**User:**  
Radio City Radio City is India's first private FM radio station and was started on 3 July 2001. It plays Hindi, English and regional songs. Radio City recently forayed into New Media in May 2008 with the launch of a music portal - PlanetRadiocity.com that offers music related news, videos, songs, and other music-related features.

**Assistant:**

```
{"named_entities": ["Radio City", "India", "3 July 2001", "Hindi", "English", "May 2008", "PlanetRadiocity.com"]}
```

#### Prompt Format

System: <system prompt>  
User: <target paragraph>

### D.1.2 Relation Extraction

#### Relation Extraction

**Goal:** Extract all underlying relations from a given paragraph.

#### System Prompt

# Role  
You are an expert in extracting all underlying relations from the given paragraph.

# Tasks  
Extract all possible underlying relations from the given [Text Chunk]. This includes both explicitly stated relations and those implicit relations that can be reasonably inferred from the [Text Chunk].  
Look for any meaningful connection between any two entities in [Entity List].

# Constraints

- No duplicate relations in your output.
- Ensure all extracted relations are concise.
- Follow exactly the format shown in the example below.

#### Task

Given a paragraph, output a JSON object containing a list of extracted relations.

#### Example

##### User:

Barack Obama was born in Hawaii. He served as the 44th President of the United States. Obama is a Democrat. He won the Nobel Peace Prize in 2009.

##### Assistant:

```
{"relations": ["was born in", "served as", "is a", "won"]}
```

#### Prompt Format

System: <system prompt>  
User:  
Text Chunk: {text\_chunk}  
Entity List: {entities}

### D.1.3 Subject-Relation Extraction

#### Subject-Relation Extraction

**Goal:** For each relation, identify which entities can serve as subjects.

#### System Prompt

# Role  
You are a Knowledge Graph Engineer.

# Tasks  
Determine which entities from the provided [Entity List] can serve as a "Subject" for each relation in the [Relation List] based on the [Text Chunk].

# Logics & Rules  
For every relation, evaluate each entity:  
- If combining the entity with the relation creates a fact supported by the text chunk (explicitly or implicitly), include that entity as a subject.  
- An entity can be a subject for multiple relations.  
- An entity does not need to be physically adjacent to the relation word in the text to be its subject.

# Constraints

- Use ONLY entities provided in the [Entity List].
- Output ONLY a JSON object where each key is a relation and the value is a list of entity names.

#### Task

Given a text chunk, a list of entities, and a list of relations, produce a mapping from each relation to plausible subject entities.

#### Prompt Format

System: <system prompt>  
User:  
Text Chunk: {text\_chunk}  
Entity List: {entities}  
Relation List: {relations}

### D.1.4 Triple Completion

#### Triple Completion

**Goal:** Complete triples by finding objects for subject-relation pairs.

#### System Prompt

# Role  
You are an expert in Knowledge Triples Completion.

# Tasks  
Complete the Knowledge Triples (Subject -> Relation -> Object) by

identifying appropriate "Objects" from the [Entity List] based on the [Text Chunk].

#### # Logics & Rules

For each (Subject, Relation) pair provided in the

[Subject-Relation Mapping]:

1. Source Check: Scan the [Text Chunk] to find any reasonable "Object"

(explicitly or implicitly reasonable) of that (Subject, Relation) pair. The "Object" MUST be an entity from the provided [Entity List].

2. Multi-Object Handling: If a (Subject, Relation) pair has multiple valid objects, separate them into multiple Dictionaries.

#### # Constraints

- Output ONLY a valid JSON array of objects.
- Each object must have exactly three keys: "subject", "relation", "object".

#### Task

Given the complete entity list and a subject-relation mapping, produce a JSON array of triples with keys subject, relation, and object.

#### Prompt Format

System: <system prompt>

User:

Text Chunk: {text\_chunk}

Entity List: {entities}

Subject-Relation Mapping: {mapping}

path needed to answer the original question.

#### # Constraints

- Start with "Seed queries:".
- Separate multiple seed queries with semicolons (;).
- Each seed query must begin with a question word, such as Who, What, When, Where, Which, How, or How many.
- Each seed query should include enough context to identify the relevant entities, but should not be overly long.
- Do not include any additional text.

#### Task

Given a complex query, output the first atomic seed query for each required reasoning path.

#### Examples

##### Example 1

**Query:** What disease did Marie Curie's husband's mother die of?

**Reasoning:** The first step is to identify Marie Curie's husband. Asking about his mother is a follow-up step.

**Output:** Seed queries: Who was Marie Curie's husband?

##### Example 2

**Query:** Were Messi and Maradona born in the same city?

**Reasoning:** The question requires two parallel first steps, one for each person's birthplace.

**Output:** Seed queries: What city was Messi born in?; What city was Maradona born in?

#### Prompt Format

System: <system prompt>

User:

Query: {query}

## D.2 Seed Query Generation

### Seed Query Generation

**Goal:** Generate the first atomic query for each reasoning path required by a complex question.

#### System Prompt

##### # Role

You are an expert at finding seed queries in complex questions.

##### # Tasks

For each complex query, identify all distinct reasoning paths required for answering it. For each reasoning path, output only the first atomic question that must be answered. Do not output follow-up questions or the original query itself.

##### # Definitions

Seed queries are simple atomic questions that can be directly answered from a knowledge base. They represent the first step in each reasoning

## D.3 Cluster Filtering

### Cluster Filtering

**Goal:** Select candidate questions that are highly relevant to a given seed query.

#### System Prompt

##### # Role

You are a relevance filter.

##### # Tasks

Given a seed query and a numbered list of candidate questions, select the numbers of the candidate questions that are highly relevant to the seed query.

##### # Relevance Criterion

A candidate question is highly relevant if it asks for information that directly helps answer the seed query, or asks for a closely related fact

or entity. If a candidate question is only loosely related or irrelevant, do not include its number.

#### # Constraints

- Output ONLY a JSON list of integers, e.g., [1, 3, 5].
- If none are relevant, output an empty list: [].
- Do not include any additional explanation.

#### Task

Given a seed query and numbered candidate questions, output the numbers of highly relevant candidates.

#### Prompt Format

System: <system prompt>  
User:  
Seed Query: {seed\_query}

Candidate Questions:  
{numbered\_list}

Output:

## D.4 Answer Sufficiency Judgement

### Answer Sufficiency Judgement

**Goal:** Determine whether the currently retrieved evidence is sufficient to answer the original question.

#### System Prompt

##### # Role

You are a QA assistant that judges whether the known information is sufficient to answer a user question.

##### # Tasks

Given the original user question and the known information retrieved so far, determine whether the question can already be answered. If the information is sufficient, provide the answer. If not, indicate that the information is insufficient.

##### # Constraints

- If the known information is sufficient, output ONLY:  
Yes. Answer: <your answer>
- If the known information is insufficient, output ONLY:  
No.
- Do not include any additional explanation.

#### Task

Given the original query and current evidence context, judge whether the evidence is sufficient and output the answer only when possible.

#### Prompt Format

System: <system prompt>

User:

User Question: {original\_query}

{context}

## D.5 Atomic Query Generation

### Atomic Query Generation

**Goal:** Generate the next atomic query based on the original question and currently retrieved evidence.

#### System Prompt

##### # Role

You are a QA assistant that decomposes complex questions into atomic follow-up queries.

##### # Tasks

Given the original user question and the known information retrieved so far, generate one new question that needs to be further queried.

##### # Constraints

- The new question must be atomic: it should ask for a single piece of information or one simple relationship.
- The new question must not be compound or multi-part.
- The new question must begin with a question word, such as Who, What, When, Where, Which, How, or How many.
- The new question should not be overly long.
- Output ONLY in the format: Query: <new question>
- Do not include any additional explanation.

#### Task

Given the original query and current evidence context, output one atomic follow-up query.

#### Prompt Format

System: <system prompt>  
User:  
User Question: {original\_query}  
  
{context}

## D.6 Dense Retrieval Fallback

### Dense Retrieval Fallback

You are a QA assistant. Based on the retrieved passages below, answer the user's question.

{context}  
User Question: {original\_query}

Output format: Answer: <your answer>  
Do not output any extra explanation.

| Benchmark       | Number of Generated Questions | Avg. Tokens / Question |
|-----------------|-------------------------------|------------------------|
| NQ              | 59,478                        | 8                      |
| PopQA           | 55,321                        | 7                      |
| HotpotQA        | 63,248                        | 13                     |
| 2WikiMultihopQA | 67,062                        | 12                     |
| MuSiQue         | 257,408                       | 15                     |

Table 7: Detailed data statistics for open-domain benchmarks. We report the number of generated questions and the average number of tokens per generated question.

| Benchmark   | Number of Generated Questions | Avg. Tokens / Question |
|-------------|-------------------------------|------------------------|
| Agriculture | 6,214                         | 8                      |
| CS          | 5,963                         | 11                     |
| Medicine    | 5,888                         | 10                     |
| Legal       | 6,123                         | 11                     |
| Mix         | 5,637                         | 9                      |

Table 8: Detailed data statistics for closed-domain benchmarks. We report the number of generated questions and the average number of tokens per generated question.

| Method                                         | NQ                       | PopQA                    | HotpotQA                 | 2WikiMultihopQA          | MuSiQue                  |
|------------------------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|
| LLM2Vec-Llama-3-8B (BehnamGhader et al., 2024) | 68.52                    | 52.65                    | 45.75                    | 47.90                    | 46.25                    |
| E5-mistral-7b-instruct (Wang et al., 2024)     | 71.04                    | 54.63                    | 47.35                    | 47.48                    | 43.25                    |
| NV-Embed-v2 (7B) (Lee et al., 2025)            | 72.22                    | 54.58                    | 51.52                    | 46.49                    | 44.59                    |
| RAPTOR (Sarthi et al., 2024)                   | 62.39                    | 52.31                    | 48.27                    | 44.88                    | 35.09                    |
| GraphRAG (Edge et al., 2024)                   | 67.82                    | 53.82                    | 45.25                    | 44.74                    | 39.76                    |
| LightRAG (Guo et al., 2025)                    | 64.23                    | 50.27                    | 42.92                    | 41.25                    | 37.01                    |
| ToG 2.0 (Ma et al., 2025)                      | 70.46                    | 55.30                    | 47.77                    | 46.92                    | 42.72                    |
| KiRAG (Fang et al., 2025)                      | 69.58                    | 53.57                    | 47.59                    | 50.31                    | 44.84                    |
| HippoRAG (Gutiérrez et al., 2024)              | 56.95                    | <u>59.12</u>             | 46.43                    | 48.99                    | 45.22                    |
| HippoRAG 2.0 (Gutiérrez et al., 2025)          | <u>75.66</u>             | 56.98                    | <u>52.43</u>             | <u>51.43</u>             | 45.66                    |
| HyperGraphRAG (Luo et al., 2026)               | 71.37                    | 56.41                    | 48.69                    | 47.97                    | 43.90                    |
| HyperRAG (Lien et al., 2026)                   | 68.74                    | 54.67                    | 46.48                    | 46.06                    | 42.05                    |
| CompactRAG (Yang et al., 2026)                 | 72.01                    | 57.13                    | 49.86                    | 48.65                    | 44.61                    |
| <b>HolisticRAG (Ours)</b>                      | <b>79.91<sup>†</sup></b> | <b>59.94<sup>†</sup></b> | <b>52.98<sup>†</sup></b> | <b>52.00<sup>†</sup></b> | <b>49.61<sup>†</sup></b> |

Table 9: Open-domain retrieval performance measured by Recall@1 using **Llama-3.3-70B-Instruct** as the LLM backbone and **NV-Embed-v2 (7B)** as the retriever. Bold and underline denote the best and second-best results, respectively. <sup>†</sup> indicates HolisticRAG significantly outperforms the best baseline under a paired Wilcoxon signed-rank test with  $p < 0.05$ .

| Method                                         | NQ                       |                          | PopQA                    |              | HotpotQA                 |              | 2WikiMultihopQA          |                          | MuSiQue                  |                          |
|------------------------------------------------|--------------------------|--------------------------|--------------------------|--------------|--------------------------|--------------|--------------------------|--------------------------|--------------------------|--------------------------|
|                                                | EM                       | F1                       | EM                       | F1           | EM                       | F1           | EM                       | F1                       | EM                       | F1                       |
| LLM2Vec-Llama-3-8B (BehnamGhader et al., 2024) | 42.84                    | 58.71                    | 32.36                    | 52.55        | 42.91                    | 66.48        | 26.54                    | 58.12                    | 29.06                    | 44.77                    |
| E5-mistral-7b-instruct (Wang et al., 2024)     | 42.57                    | 56.72                    | 31.05                    | 55.39        | 50.64                    | 69.41        | 27.02                    | 58.84                    | 25.62                    | 46.91                    |
| NV-Embed-v2 (7B) (Lee et al., 2025)            | 41.93                    | 60.06                    | 31.88                    | 54.42        | 48.77                    | 71.55        | 23.91                    | 58.63                    | 30.21                    | 43.86                    |
| RAPTOR (Sarathi et al., 2024)                  | 44.10                    | 48.76                    | 31.61                    | 54.52        | 47.83                    | 65.62        | 35.87                    | 51.91                    | 31.28                    | 27.84                    |
| GraphRAG (Edge et al., 2024)                   | 40.87                    | 46.64                    | 26.92                    | 50.73        | 49.95                    | 64.82        | 34.09                    | 55.88                    | 23.91                    | 36.74                    |
| LightRAG (Guo et al., 2025)                    | 4.58                     | 21.41                    | 2.43                     | 5.86         | 2.06                     | 4.61         | 2.78                     | 9.93                     | 1.84                     | 3.72                     |
| ToG 2.0 (Ma et al., 2025)                      | 44.63                    | 51.21                    | 30.54                    | 53.44        | 52.51                    | 58.72        | 32.91                    | 49.96                    | 31.72                    | 45.08                    |
| KiRAG (Fang et al., 2025)                      | 47.74                    | 55.36                    | 28.58                    | 51.69        | 47.63                    | 64.20        | 33.69                    | 56.17                    | 24.71                    | 32.05                    |
| HippoRAG (Gutiérrez et al., 2024)              | 42.69                    | 52.74                    | 23.86                    | 51.72        | 53.75                    | 65.13        | 39.51                    | 46.83                    | 25.16                    | 35.61                    |
| HippoRAG 2.0 (Gutiérrez et al., 2025)          | 48.31                    | 61.64                    | 32.88                    | 50.31        | 57.58                    | <b>74.35</b> | 39.77                    | 48.95                    | 26.62                    | 40.06                    |
| HyperRAG (Lien et al., 2026)                   | 47.96                    | 60.22                    | 32.94                    | 52.18        | 56.41                    | 69.84        | 40.67                    | 56.31                    | 32.73                    | 47.58                    |
| HyperGraphRAG (Luo et al., 2026)               | 49.38                    | 61.57                    | <u>34.62</u>             | 53.86        | <u>57.82</u>             | 71.36        | <u>42.31</u>             | <u>57.94</u>             | <u>34.86</u>             | <u>49.27</u>             |
| CompactRAG (Yang et al., 2026)                 | 44.72                    | 58.05                    | 30.02                    | <b>57.42</b> | 49.83                    | 61.04        | 25.16                    | 28.01                    | 24.05                    | 35.17                    |
| <b>HolisticRAG (Ours)</b>                      | <b>52.04<sup>†</sup></b> | <b>64.86<sup>†</sup></b> | <b>37.08<sup>†</sup></b> | <u>55.71</u> | <b>60.46<sup>†</sup></b> | <u>74.02</u> | <b>45.29<sup>†</sup></b> | <b>60.88<sup>†</sup></b> | <b>38.42<sup>†</sup></b> | <b>53.11<sup>†</sup></b> |

Table 10: Open-domain QA performance using **Llama-3.3-70B-Instruct** as the LLM backbone and **NV-Embed-v2 (7B)** as the retriever. For HyperRAG, we report the better result between its HyperRetriever and HyperMemory settings. Bold and underline denote the best and second-best results, respectively. <sup>†</sup> indicates HolisticRAG significantly outperforms the best baseline under a paired Wilcoxon signed-rank test with  $p < 0.05$ .

| Method                           | Agriculture              |              | CS                       |                          | Legal        |              | Mix                      |                          | Medicine     |                          |
|----------------------------------|--------------------------|--------------|--------------------------|--------------------------|--------------|--------------|--------------------------|--------------------------|--------------|--------------------------|
|                                  | EM                       | F1           | EM                       | F1                       | EM           | F1           | EM                       | F1                       | EM           | F1                       |
| HyperGraphRAG (Luo et al., 2026) | 15.89                    | 31.64        | 17.77                    | 32.32                    | <b>21.55</b> | <b>43.32</b> | 29.88                    | 39.87                    | <b>22.34</b> | 35.49                    |
| <b>HolisticRAG (Ours)</b>        | <b>16.33<sup>†</sup></b> | <b>31.88</b> | <b>18.89<sup>†</sup></b> | <b>34.67<sup>†</sup></b> | 20.33        | 39.58        | <b>30.52<sup>†</sup></b> | <b>40.54<sup>†</sup></b> | 20.21        | <b>37.82<sup>†</sup></b> |

Table 11: Closed-domain QA performance on five benchmarks using **Llama-3.3-70B-Instruct** as the LLM backbone and **NV-Embed-v2 (7B)** as the retriever, comparing HolisticRAG with HyperGraphRAG. Bold and underline denote the best and second-best results, respectively. <sup>†</sup> indicates HolisticRAG significantly outperforms the best baseline under a paired Wilcoxon signed-rank test with  $p < 0.05$ .

| Method                                         | NQ                       | PopQA                    | HotpotQA                 | 2WikiMultihopQA          | MuSiQue                  |
|------------------------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|
| LLM2Vec-Llama-3-8B (BehnamGhader et al., 2024) | 69.75                    | 54.21                    | 46.64                    | 48.95                    | <u>47.44</u>             |
| E5-mistral-7b-instruct (Wang et al., 2024)     | 72.12                    | 55.57                    | 48.91                    | 49.15                    | 44.70                    |
| NV-Embed-v2 (7B) (Lee et al., 2025)            | 73.53                    | 56.22                    | 52.49                    | 47.92                    | 45.97                    |
| RAPTOR (Sarathi et al., 2024)                  | 64.11                    | 53.84                    | 49.22                    | 46.49                    | 36.58                    |
| GraphRAG (Edge et al., 2024)                   | 69.16                    | 55.17                    | 46.43                    | 46.15                    | 41.21                    |
| LightRAG (Guo et al., 2025)                    | 65.57                    | 51.61                    | 44.10                    | 42.66                    | 38.46                    |
| ToG 2.0 (Ma et al., 2025)                      | 71.80                    | 56.64                    | 48.95                    | 48.33                    | 44.17                    |
| KiRAG (Fang et al., 2025)                      | 70.84                    | 54.49                    | 48.93                    | 52.00                    | 46.39                    |
| HippoRAG (Gutiérrez et al., 2024)              | 58.36                    | <u>59.99</u>             | 47.95                    | 50.35                    | 46.96                    |
| HippoRAG 2.0 (Gutiérrez et al., 2025)          | <u>76.91</u>             | 58.60                    | <u>53.54</u>             | <u>52.31</u>             | 46.99                    |
| HyperGraphRAG (Luo et al., 2026)               | 72.71                    | 57.76                    | 49.87                    | 49.38                    | 45.35                    |
| HyperRAG (Lien et al., 2026)                   | 70.08                    | 56.02                    | 47.66                    | 47.47                    | 43.50                    |
| CompactRAG (Yang et al., 2026)                 | 73.35                    | 58.48                    | 51.04                    | 50.06                    | 46.06                    |
| <b>HolisticRAG (Ours)</b>                      | <b>81.35<sup>†</sup></b> | <b>61.62<sup>†</sup></b> | <b>54.04<sup>†</sup></b> | <b>53.57<sup>†</sup></b> | <b>51.10<sup>†</sup></b> |

Table 12: Open-domain retrieval performance measured by Recall@1 using **GPT-4o** as the LLM backbone and **NV-Embed-v2 (7B)** as the retriever. Bold and underline denote the best and second-best results, respectively. <sup>†</sup> indicates HolisticRAG significantly outperforms the best baseline under a paired Wilcoxon signed-rank test with  $p < 0.05$ .

| Method                                         | NQ                       |                          | PopQA                    |                          | HotpotQA                 |                          | 2WikiMultihopQA          |                          | MuSiQue                  |                          |
|------------------------------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|--------------------------|
|                                                | EM                       | F1                       | EM                       | F1                       | EM                       | F1                       | EM                       | F1                       | EM                       | F1                       |
| LLM2Vec-Llama-3-8B (BehnamGhader et al., 2024) | 44.18                    | 60.09                    | 33.81                    | 53.94                    | 44.22                    | 67.96                    | 28.06                    | 59.54                    | 30.47                    | 46.08                    |
| E5-mistral-7b-instruct (Wang et al., 2024)     | 44.03                    | 58.18                    | 32.37                    | 56.82                    | 52.07                    | 70.86                    | 28.49                    | 60.22                    | 27.04                    | 48.37                    |
| NV-Embed-v2 (7B) (Lee et al., 2025)            | 43.41                    | 61.49                    | 33.16                    | 55.83                    | 50.13                    | 72.97                    | 25.24                    | 60.08                    | 31.72                    | 45.39                    |
| RAPTOR (Sarifi et al., 2024)                   | 45.72                    | 50.18                    | 33.05                    | 55.89                    | 49.16                    | 67.01                    | 37.34                    | 53.26                    | 32.61                    | 29.22                    |
| GraphRAG (Edge et al., 2024)                   | 42.34                    | 48.03                    | 28.29                    | 52.15                    | 51.28                    | 66.19                    | 35.56                    | 57.42                    | 25.27                    | 38.06                    |
| LightRAG (Guo et al., 2025)                    | 5.91                     | 22.86                    | 3.71                     | 7.12                     | 3.34                     | 5.94                     | 4.02                     | 11.36                    | 3.18                     | 5.06                     |
| ToG 2.0 (Ma et al., 2025)                      | 46.02                    | 52.70                    | 31.88                    | 54.92                    | 53.94                    | 60.13                    | 34.36                    | 51.37                    | 33.14                    | 46.49                    |
| KiRAG (Fang et al., 2025)                      | 49.18                    | 56.81                    | 30.03                    | 53.02                    | 49.11                    | 65.66                    | 35.12                    | 57.58                    | 26.03                    | 33.48                    |
| HippoRAG (Gutiérrez et al., 2024)              | 44.17                    | 54.21                    | 25.21                    | 53.07                    | 55.16                    | 66.59                    | 40.96                    | 48.31                    | 26.57                    | 37.04                    |
| HippoRAG 2.0 (Gutiérrez et al., 2025)          | 49.74                    | 63.12                    | 34.25                    | 51.78                    | 58.93                    | 71.54                    | 41.16                    | 50.42                    | 28.01                    | 41.48                    |
| HyperRAG (Lien et al., 2026)                   | 50.87                    | 63.74                    | 35.84                    | 55.19                    | 59.21                    | 72.13                    | 43.96                    | 59.68                    | 36.97                    | 51.76                    |
| HyperGraphRAG (Luo et al., 2026)               | <u>52.46</u>             | <u>65.08</u>             | <u>37.21</u>             | 56.71                    | <u>60.68</u>             | <u>73.92</u>             | <u>45.37</u>             | <u>61.34</u>             | <u>38.62</u>             | <u>53.48</u>             |
| CompactRAG (Yang et al., 2026)                 | 46.09                    | 59.52                    | 31.38                    | <u>56.84</u>             | 51.28                    | 62.43                    | 26.58                    | 29.47                    | 25.44                    | 36.62                    |
| <b>HolisticRAG (Ours)</b>                      | <b>53.71<sup>†</sup></b> | <b>66.42<sup>†</sup></b> | <b>38.66<sup>†</sup></b> | <b>57.23<sup>†</sup></b> | <b>61.94<sup>†</sup></b> | <b>75.41<sup>†</sup></b> | <b>46.86<sup>†</sup></b> | <b>62.43<sup>†</sup></b> | <b>40.05<sup>†</sup></b> | <b>54.74<sup>†</sup></b> |

Table 13: Open-domain QA performance using **GPT-4o** as the LLM backbone and **NV-Embed-v2 (7B)** as the retriever. For HyperRAG, we report the better result between its HyperRetriever and HyperMemory settings. Bold and underline denote the best and second-best results, respectively. <sup>†</sup> indicates HolisticRAG significantly outperforms the best baseline under a paired Wilcoxon signed-rank test with  $p < 0.05$ .

| Method                                  | Agriculture              |                          | CS           |                          | Legal        |                          | Mix          |              | Medicine     |                          |
|-----------------------------------------|--------------------------|--------------------------|--------------|--------------------------|--------------|--------------------------|--------------|--------------|--------------|--------------------------|
|                                         | EM                       | F1                       | EM           | F1                       | EM           | F1                       | EM           | F1           | EM           | F1                       |
| <b>HyperGraphRAG (Luo et al., 2026)</b> | 17.03                    | 31.71                    | <b>23.66</b> | 31.27                    | <b>28.54</b> | 43.88                    | <b>32.08</b> | <b>45.12</b> | <b>27.03</b> | 39.89                    |
| HolisticRAG (Ours)                      | <b>18.62<sup>†</sup></b> | <b>33.67<sup>†</sup></b> | 19.68        | <b>33.59<sup>†</sup></b> | 24.64        | <b>45.33<sup>†</sup></b> | 29.53        | 43.10        | 24.64        | <b>40.37<sup>†</sup></b> |

Table 14: Closed-domain QA performance on five benchmarks using **GPT-4o** as the LLM backbone and **NV-Embed-v2 (7B)** as the retriever, comparing HolisticRAG with HyperGraphRAG. Bold and underline denote the best and second-best results, respectively. <sup>†</sup> indicates HolisticRAG significantly outperforms the best baseline under a paired Wilcoxon signed-rank test with  $p < 0.05$ .

| Method                    | Offline Time / Chunk (s) | Offline Tokens / Chunk | Online Time / Query (s) | Online Tokens / Query |
|---------------------------|--------------------------|------------------------|-------------------------|-----------------------|
| RAPTOR                    | 5.33                     | 1,533                  | 3.72                    | 1,098                 |
| GraphRAG                  | 7.32                     | 1,598                  | 2.33                    | 1,207                 |
| LightRAG                  | 5.69                     | 1,459                  | 2.57                    | 1,037                 |
| ToG 2.0                   | —                        | —                      | 4.52                    | 1,968                 |
| KiRAG                     | 7.33                     | 1,539                  | 4.78                    | 1,688                 |
| HippoRAG 2.0              | 6.13                     | 1,068                  | 1.37                    | 483                   |
| CompactRAG                | 6.77                     | 1,453                  | 3.24                    | 982                   |
| HyperGraphRAG             | 5.47                     | 1,256                  | 1.68                    | 569                   |
| HyperRAG                  | 5.47                     | 1,256                  | 1.92                    | 513                   |
| <b>HolisticRAG (Ours)</b> | 7.10                     | 1,478                  | 2.78                    | 1,457                 |

Table 15: Cost and efficiency comparison on HotpotQA using **Qwen2.5-32B-Instruct** as the LLM backbone and **NV-Embed-v2 (7B)** as the retriever. Lower values indicate better efficiency. “—” indicates that the method does not require an offline LLM-based indexing stage under its original setting.

| Method                    | Offline Time / Chunk (s) | Offline Tokens / Chunk | Online Time / Query (s) | Online Tokens / Query |
|---------------------------|--------------------------|------------------------|-------------------------|-----------------------|
| RAPTOR                    | 6.94                     | 1,548                  | 4.84                    | 1,115                 |
| GraphRAG                  | 9.52                     | 1,621                  | 3.03                    | 1,226                 |
| LightRAG                  | 7.40                     | 1,472                  | 3.34                    | 1,051                 |
| ToG 2.0                   | –                        | –                      | 5.88                    | 1,994                 |
| KiRAG                     | 9.53                     | 1,557                  | 6.21                    | 1,711                 |
| HippoRAG 2.0              | 7.97                     | 1,081                  | 1.78                    | 491                   |
| CompactRAG                | 8.80                     | 1,470                  | 4.21                    | 995                   |
| HyperGraphRAG             | 7.11                     | 1,273                  | 2.18                    | 578                   |
| HyperRAG                  | 7.11                     | 1,271                  | 2.50                    | 521                   |
| <b>HolisticRAG (Ours)</b> | 9.23                     | 1,494                  | 3.61                    | 1,476                 |

Table 16: Cost and efficiency comparison on HotpotQA using **Llama-3.3-70B-Instruct** as the LLM backbone and **NV-Embed-v2 (7B)** as the retriever. Lower values indicate better efficiency. “–” indicates that the method does not require an offline LLM-based indexing stage under its original setting.

| Method                    | Offline Time / Chunk (s) | Offline Tokens / Chunk | Online Time / Query (s) | Online Tokens / Query |
|---------------------------|--------------------------|------------------------|-------------------------|-----------------------|
| RAPTOR                    | 5.18                     | 1,526                  | 3.61                    | 1,087                 |
| GraphRAG                  | 7.11                     | 1,607                  | 2.26                    | 1,194                 |
| LightRAG                  | 5.52                     | 1,451                  | 2.49                    | 1,029                 |
| ToG 2.0                   | –                        | –                      | 4.39                    | 1,951                 |
| KiRAG                     | 7.12                     | 1,548                  | 4.64                    | 1,671                 |
| HippoRAG 2.0              | 5.95                     | 1,061                  | 1.33                    | 479                   |
| CompactRAG                | 6.57                     | 1,444                  | 3.14                    | 974                   |
| HyperGraphRAG             | 5.31                     | 1,248                  | 1.63                    | 563                   |
| HyperRAG                  | 5.31                     | 1,249                  | 1.86                    | 508                   |
| <b>HolisticRAG (Ours)</b> | 6.89                     | 1,469                  | 2.70                    | 1,444                 |

Table 17: Cost and efficiency comparison on HotpotQA using **GPT-4o** as the LLM backbone and **NV-Embed-v2 (7B)** as the retriever. Lower values indicate better efficiency. “–” indicates that the method does not require an offline LLM-based indexing stage under its original setting.

| Number of Candidate Questions | HotpotQA F1  |
|-------------------------------|--------------|
| 10                            | 71.68        |
| 15                            | 71.97        |
| <b>20</b>                     | <b>72.63</b> |
| 25                            | 70.34        |
| 30                            | 69.12        |

Table 18: Effect of the number of retrieved candidate questions before graph-grounded clustering on HotpotQA using **Qwen2.5-32B-Instruct** as the LLM backbone and **NV-Embed-v2 (7B)** as the retriever. Bold values indicate the best performance.

| Maximum Inference Depth | HotpotQA F1  |
|-------------------------|--------------|
| 1                       | 51.32        |
| 2                       | 63.32        |
| 3                       | 69.38        |
| 4                       | 71.89        |
| <b>5</b>                | <b>72.63</b> |
| 6                       | 70.39        |
| 7                       | 68.55        |
| 8                       | 62.61        |

Table 19: Effect of the maximum online inference depth on HotpotQA using **Qwen2.5-32B-Instruct** as the LLM backbone and **NV-Embed-v2 (7B)** as the retriever. Bold values indicate the best performance.

| Fallback Threshold | HotpotQA F1  |
|--------------------|--------------|
| 0.95               | 72.46        |
| <b>0.90</b>        | <b>72.63</b> |
| 0.85               | 71.79        |
| 0.80               | 71.81        |

Table 20: Effect of the fallback threshold on HotpotQA using **Qwen2.5-32B-Instruct** as the LLM backbone and **NV-Embed-v2 (7B)** as the retriever. Bold values indicate the best performance.

| #  | Original Query                                                                                                                                                                    | Generated Seed Queries                                                                                                                                     |
|----|-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------|------------------------------------------------------------------------------------------------------------------------------------------------------------|
| 1  | The fictional private detective that appears in “The Adventure of the Seven Clocks” what written by whom?                                                                         | Who wrote “The Adventure of the Seven Clocks”?                                                                                                             |
| 2  | The Distribution of Industry act was passed by a man who was prime minister when?                                                                                                 | When was the man who passed the Distribution of Industry act prime minister?                                                                               |
| 3  | Sparking the Marian civil war, who helped the recently abdicated queen to escape her imprisonment?                                                                                | Who was the recently abdicated queen in the Marian civil war?<br>Who helped the queen escape her imprisonment?                                             |
| 4  | What screenplay was worked on by both Edward Carfagno and Miklos Rozsa?                                                                                                           | What screenplays did Edward Carfagno work on?<br>What screenplays did Miklos Rozsa work on?                                                                |
| 5  | Are Colocasia and Coronilla both flowering plants?                                                                                                                                | Are Colocasia flowering plants?<br>Are Coronilla flowering plants?                                                                                         |
| 6  | Which is located in Cornwall, Southwell Minster or Truro Cathedral?                                                                                                               | Where is Southwell Minster located?<br>Where is Truro Cathedral located?                                                                                   |
| 7  | How do the Peter Laufer books Forbidden Creatures and No Animals Were Harmed differ in their focus on animals?                                                                    | What is the main focus of Peter Laufer’s book Forbidden Creatures?<br>What is the main focus of Peter Laufer’s book No Animals Were Harmed?                |
| 8  | Are Steve Perry and Leslie West both singers?                                                                                                                                     | Is Steve Perry a singer?<br>Is Leslie West a singer?                                                                                                       |
| 9  | In what year was the Golden State NBA player, who was part of the Cavaliers-Warriors rivalry, named NBA Finals Most Valuable Player?                                              | Who was the Golden State NBA player part of the Cavaliers-Warriors rivalry?<br>When was the Golden State NBA player named NBA Finals Most Valuable Player? |
| 10 | What is the name of the actor who wrote the memoir Without you: A memoir of love, loss, and the Musical Rent and also starred as Mark Cohen in the Broadway production of “Rent”? | Who wrote the memoir Without you: A memoir of love, loss, and the Musical Rent?<br>Who played Mark Cohen in the Broadway production of “Rent”?             |

Table 21: Examples of seed queries generated by Qwen2.5-32B-Instruct for ten complex queries sampled from HotpotQA.