Text-Graph Synergy: A Bidirectional Verification and Completion Framework for RAG

Jiarui Zhong , Hong Cai Chen*
School of Automation, Southeast University, Nanjing 210096, China
220245143@seu.edu.cn, chenhc@seu.edu.cn

Abstract

Retrieval-Augmented Generation (RAG) has become a core paradigm for enhancing factual grounding and multi-hop reasoning in Large Language Models (LLMs). Traditional text-based RAG often retrieves logically irrelevant pseudo-evidence, while graph-based RAG is frequently hindered by search-time pruning, which may discard potentially valid reasoning paths. Existing hybrid approaches primarily adopt simple evidence concatenation or unidirectional enhancement, which fails to address the fundamental "Information Island" problem caused by asymmetric reasoning flows between unstructured text and structured graphs. We propose **TGS-RAG**, a unified framework for **T**ext-**G**raph **S**ynergistic enhancement. TGS-RAG introduces a bidirectional mechanism: (i) a **Graph-to-Text** channel that employs a Global Voting strategy from visited graph nodes to re-rank and refine textual evidence, filtering out semantic noise; and (ii) a **Text-to-Graph** channel that utilizes the **Memory-based Orphan Entity Bridging** algorithm. This algorithm utilizes textual cues to proactively resurrect valid but previously pruned reasoning paths from the search history without additional database overhead. Experimental results on multiple multi-hop reasoning benchmarks demonstrate that TGS-RAG significantly outperforms state-of-the-art baselines, achieving a superior balance between retrieval precision and computational efficiency.

1 Introduction

Large Language Models (LLMs) have achieved remarkable performance in natural language understanding and generation across a wide range of tasks. However, despite their impressive parametric capacity, LLMs remain fundamentally limited by their reliance on static internal knowledge and their tendency to produce hallucinated or logically inconsistent responses [Ji *et al.*, 2023], particularly in knowledge-intensive and multi-hop reasoning scenarios. Retrieval-Augmented Generation (RAG) has therefore emerged as a central paradigm for grounding LLM outputs in external, verifiable evidence, enabling more factual, controllable, and interpretable generation [Lewis *et al.*, 2020].

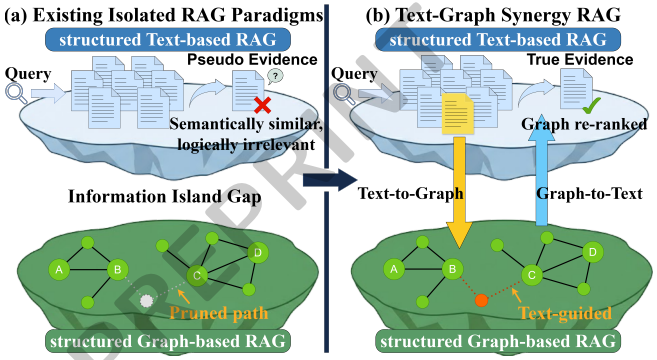

Figure 1: Comparison between isolated retrieval paradigms and the TGS-RAG framework. (a) Existing paradigms suffer from the "Information Island" gap: text-based methods often retrieve semantically similar but logically irrelevant *pseudo-evidence*, while graph-based methods are hindered by *broken reasoning paths* due to search-time pruning. (b) TGS-RAG bridges this gap through a bidirectional synergy mechanism, where graph structure guides the re-ranking of text to identify true evidence, and textual context facilitates the recovery of pruned but potentially valid reasoning paths.

Existing RAG systems predominantly follow two distinct paradigms. **Text-based RAG** retrieves relevant information from large unstructured corpora using dense vector similarity. While this paradigm offers high coverage, it is prone to retrieving pseudo-evidence [Cossio, 2025]—textual chunks that are semantically similar to the query but logically irrelevant, and struggle with complex multi-hop reasoning. In contrast, **Graph-based RAG** leverages structured Knowledge Graphs (KGs) to provide high logical interpretability. However, these systems are often constrained by both structural sparsity in KGs and search-time pruning that may discard potentially useful reasoning paths.

Recent research attempts to combine Text-based RAG and Graph-based RAG to exploit their complementary strengths. However, these approaches often rely on simple evidence concatenation or pipeline-style augmentation, which treat textual and graphical evidence as independent sources, fail-

*Corresponding author.

ing to achieve deep integration. Furthermore, unidirectional enhancement frameworks (e.g., KG-infused RAG) typically utilize one modality merely as an auxiliary signal for the other. Such designs fail to establish a closed-loop interaction, leaving the deeper potential of mutual verification and co-discovery largely unexplored.

At the core of this limitation lies the **Information Island problem**: textual evidence and graphical evidence are retrieved and processed in isolation, without a mechanism for mutual validation or collaborative reasoning. Specifically, current systems lack a principled framework in which (i) the structured logical constraints encoded in graphs can guide and refine text retrieval, while simultaneously (ii) the rich contextual cues present in unstructured text can validate, enrich, and recover pruned but potentially useful graph reasoning paths. In practice, this often means that the text and graph channels retrieve evidence that appears relevant on their own, but fails to mutually support a coherent reasoning chain.

To address these challenges, we propose **Text-Graph Synergy RAG (TGS-RAG)**[1], a novel framework that enables deep, bidirectional integration between unstructured text and structured knowledge graphs during retrieval. TGS-RAG introduces a bidirectional enhancement mechanism: a *Graph-to-Text* channel that utilizes structured logic to re-rank textual chunks, and a *Text-to-Graph* channel that leverages contextual clues to validate and complete KG paths. Our main contributions are as follows:

- **Bidirectional Synergy Framework:** We propose TGS-RAG, a unified framework that breaks the "information island" barrier by establishing a closed-loop feedback mechanism: graph structures guide the re-ranking of textual evidence (Graph-to-Text), while textual context validates and repairs graph reasoning paths (Text-to-Graph). This closed-loop design allows the two channels to move beyond simple coexistence and mutually verify the evidence retrieved by each other.
- **Memory-based Orphan Entity Bridging:** We introduce a novel algorithm that addresses the loss of potentially useful reasoning paths during beam-search pruning by treating pruned nodes as a *deferred reasoning memory*. By leveraging textual cues to "resurrect" these orphan entities, we recover potentially valid reasoning paths discarded during initial search.
- **Cost-Effective Dual-Channel Reasoning:** We design a synergistic scoring mechanism that combines semantic similarity with structural voting. This approach effectively filters out pseudo-evidence and achieves an optimal Pareto frontier, delivering state-of-the-art accuracy with significantly lower token consumption than global graph indexing methods.
- **Empirical Superiority:** Extensive experiments on MuSiQue and HotpotQA demonstrate that TGS-RAG significantly outperforms existing text-based, graph-based, and hybrid baselines, particularly in scenarios requiring complex multi-hop reasoning across disjointed evidence.

[1]https://github.com/EvannZhongg/TGS_RAG.git

2 Related Work

The evolution of RAG has transitioned from simple text matching to complex multi-modal and structural fusion. Our work builds upon three main research areas.

2.1 Unstructured Retrieval for RAG

Traditional RAG systems primarily rely on unstructured retrieval, where dense vector representations facilitate semantic matching between queries and documents. Dense Passage Retrieval (DPR) [Karpukhin *et al.*, 2020] pioneered the dual-encoder architecture for open-domain question answering, significantly improving recall over lexical methods. Subsequent models like ColBERT [Khattab and Zaharia, 2020] introduced late interaction mechanisms to capture fine-grained token-level similarities. While these models excel at broad knowledge coverage, they often struggle with multi-hop queries where semantic similarity does not necessarily imply logical relevance, leading to the retrieval of misleading "pseudo-evidence."

2.2 Structured Retrieval for RAG

To address the logical limitations of text-based retrieval, recent studies have explored Knowledge Graphs (KGs) as structured knowledge sources. Methods in this category typically extract subgraphs or relational paths to provide deterministic evidence for LLMs. For instance, **G-Retriever** [He *et al.*, 2024] employs a GNN-based adapter to filter irrelevant nodes and edges, targeting efficient subgraph retrieval for textual graph understanding. Similarly, KG-based reasoning frameworks like ToG [Sun *et al.*, 2024] utilize LLMs as agents to execute path-searching algorithms (e.g., beam search) over structured facts. Beyond path reasoning, recent graph-based indexing approaches leverage LLM-generated structures to capture global information. **GraphRAG** [Edge *et al.*, 2025] constructs hierarchical community summaries to answer global queries but suffers from high indexing costs. In contrast, **LightRAG** [Guo *et al.*, 2025] introduces a dual-level retrieval paradigm incorporating both graph structures and vector representations. However, these indexing-focused approaches primarily address retrieval efficiency and coverage, but are less equipped to handle scenarios where potentially useful reasoning paths are discarded during search.

2.3 Text-Graph Integration Paradigms

The integration of text and graphs has become a promising direction for robust RAG systems. Early approaches typically employed unidirectional enhancement strategies. For instance, KG-Infused RAG utilizes graph entities to expand queries for text retrieval, while frameworks like KG^2RAG [Zhu *et al.*, 2025] adopt a "Semantic-to-Graph" paradigm, using text seeds from initial semantic retrieval to expand graph components. However, these methods often face a "unidirectional bottleneck," where the strengths of one modality do not actively feed back to correct or complete the other in real-time.

To overcome these limitations, recent frameworks have moved toward cross-source validation. **Think-on-Graph 2.0 (ToG-2)** [Ma *et al.*, 2025] introduces a tight-coupling

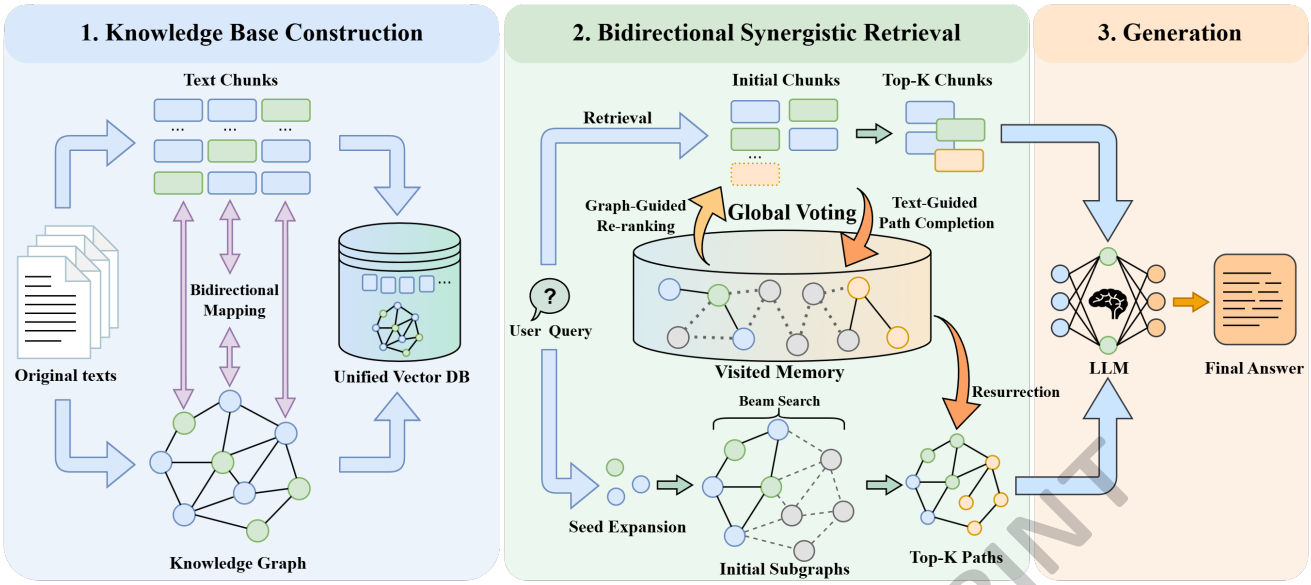

Figure 2: The overall architecture of TGS-RAG. The framework operates in three phases: (1) **Knowledge Base Construction**, where text chunks and entities are mapped in a unified vector DB; (2) **Bidirectional Synergistic Retrieval**, the core stage utilizing *Semantic Beam Search* to accumulate a **Visited Memory** containing both selected and pruned nodes. This memory facilitates *Graph-Guided Re-ranking* via **Global Voting** and enables *Text-Guided Path Completion* by **resurrecting** pruned orphan entities from memory to bridge logical gaps; and (3) **Generation**, synthesizing the refined evidence.

paradigm that iteratively alternates between KG-based graph retrieval and document-based context retrieval, using text evidence to prune graph paths and graph structures to guide text retrieval. While TGS-RAG shares the high-level goal of mutual verification, the underlying mechanisms differ substantially. ToG-2 follows a subtractive strategy, repeatedly pruning and re-expanding the search space through costly iterative retrieval. In contrast, TGS-RAG adopts an additive strategy via Memory-based Orphan Entity Bridging, which treats pruning as postponement rather than elimination and directly resurrects pruned paths from the semantic beam search history. This retrospective repair mechanism enables efficient recovery of logical chains that are difficult to preserve under sparse graph connectivity and search-time pruning without additional database queries.

3 The TGS-RAG Framework

As illustrated in Figure 2, TGS-RAG is designed as a two-stage framework that tightly integrates unstructured text and structured knowledge graphs for retrieval-augmented reasoning. It consists of an offline **Knowledge Base Construction** stage, responsible for extracting and fusing knowledge from unstructured documents; and an online **Bidirectional Synergistic Retrieval** stage, which performs mutual verification and enhancement between text and graph evidence during inference.

3.1 Knowledge Base Construction

Given an unstructured document corpus C, TGS-RAG constructs a unified knowledge base through an offline processing pipeline. We leverage Large Language Models (LLMs) to extract salient entities E and their semantic relations R, forming a global knowledge graph $G = (E, R)$.

Crucially, we establish a **bidirectional mapping** M between the text corpus C and the knowledge graph G. Each relational triplet in G is explicitly linked to its originating text chunks, while each text chunk is annotated with the entities and relations it contains. This mapping transforms the knowledge graph from a standalone structure into a reasoning interface over text. Formally, the knowledge base is defined as a triplet:

$$KB = (C, G, M) \tag{1}$$

where M represents the bidirectional links between C and G. This unified and interconnected knowledge base serves as the foundation for the proposed bidirectional enhanced retrieval framework.

3.2 Dual-Channel Initial Retrieval

Given a user query q, TGS-RAG first extracts a set of query-relevant entity mentions using an LLM. These entities are embedded to obtain vectors v_E, which are then used to retrieve a seed entity set E_{seed} from the knowledge graph G.

Starting from E_{seed}, TGS-RAG performs two complementary retrieval processes in parallel: a Text Channel and a Graph Channel. This dual-channel design intentionally captures heterogeneous evidence signals that exhibit different failure modes.

- **Text Channel**: This channel performs standard semantic retrieval to quickly recall textual evidence directly related to the user query. We use the query vector v_q of user query q to perform vector indexing on the text

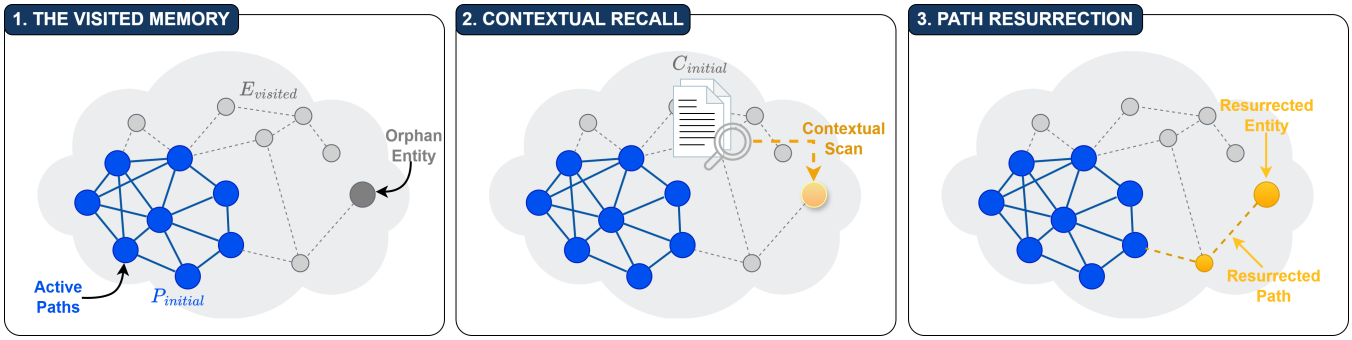

Figure 3: Illustration of the Memory-based Orphan Entity Bridging algorithm. The process operates in three steps: (1) **The Visited Memory**: Semantic Beam Search generates a set of *Active Paths* ($P_{initial}$, blue nodes) while implicitly storing pruned nodes in the *Visited Memory* ($E_{visited}$, grey area); (2) **Contextual Recall**: Contextual clues from retrieved text chunks ($C_{initial}$) perform a **Contextual Scan** to identify relevant but pruned "orphan entities"; (3) **Path Resurrection**: The pruning decision is reversed, and the *Resurrected Path* (p_{bridge}, orange) connecting the resurrected entity to the main subgraph is recovered without additional database queries.

chunk collection C, retrieving the initial text set $C_{initial}$ with the highest similarity to v_q.

- **Graph Channel**: Starting from the seed entity set E_{seed}, the Semantic Beam Search algorithm computes, at each hop, the cosine similarity between the embeddings of neighboring nodes and the query vector v_q. Only the top-K (Beam Width) most semantically relevant paths are retained for the next expansion step. This heuristic pruning ensures that the retrieval focuses on the semantic neighborhood of the query, yielding a high-quality path set $P_{initial}$ within a preset depth d.

The outcome of this stage is two heterogeneous evidence sets: C_{initial}, which emphasizes semantic recall, and P_{initial}, which captures structured reasoning paths.

3.3 Synergistic Fusion and Enhancement

After obtaining the initial text evidence $C_{initial}$ and graph path set $P_{initial}$, the TGS-RAG introduces a bidirectional scoring and discovery mechanism that enables mutual verification and refinement of information. This process includes two enhancement directions: *Graph-to-Text* and *Text-to-Graph*.

Graph-to-Text Synergy

This process utilizes structured knowledge from the graph channel to guide the rediscovery and re-ranking of text evidence, rather than relying only on surface-level similarity to the query.

1. **Chunk Recommendation**: We aggregate all entities visited during the Semantic Beam Search (including those in pruned paths) to form a global visited set $E_{visited}$. These entities act as "recommenders" to "vote" for their source text chunks. The graph-recommended chunks are merged with $C_{initial}$ to create a broader candidate pool $C_{candidate}$, allowing text chunks that have low semantic similarity but are structurally relevant to be recalled.
2. **Chunk Re-ranking**: We calculate a recommendation score $Rec(c)$ for each text chunk $c \in C_{candidate}$, proportional to the number of entities recommending it. The final score $Score_{final}(c)$ is a weighted fusion of its original semantic similarity and the graph recommendation score:

$$Score_{\text{final}}(c) = \alpha \cdot \text{Norm}\big(\text{sim}(v_q, v_c)\big) + (1-\alpha) \cdot \text{Norm}\big(\text{Rec}(c)\big) \quad (2)$$

where α is a hyperparameter balancing the two scores.

Text-to-Graph Synergy

This process utilizes the rich contextual information in the text channel to verify and enhance the structured paths discovered in the graph channel. As illustrated in Figure 3, this involves confirming existing paths and resurrecting pruned connections.

For each path $p \in P_{initial}$, we first calculate a base quality score $Score_{base}(p)$ based on the semantic similarity of its components to the query q and their structural importance.

1. **Path Confirmation**: The path is validated by checking the intersection of entities on the path and entities contained in $C_{initial}$. If a path receives support from text evidence (i.e., its entities appear in the retrieved text chunks), its score is boosted:

$$Score_{conf}(p) = Score_{base}(p) + \epsilon \cdot |Entities(p) \cap Entities(C_{initial})| \quad (3)$$

where ϵ is the weight for text confirmation reward.

2. **Memory-based Orphan Entity Bridging**: To discover potential knowledge ignored by the initial path pruning, we define **Orphan Entities** E_{orphan} as key entities that are activated by the retrieved text evidence $C_{initial}$ but are absent from the initially selected graph paths $P_{initial}$. Instead of performing a costly new search or re-expansion, we leverage the **Visited Memory** as a *deferred reasoning buffer*. If an orphan entity exists in the visited but pruned nodes, the original pruning decision is reversed by directly replaying its stored path, thereby resurrecting a valid reasoning chain without additional database queries. We check if any orphan entity exists in the set of visited nodes that were pruned during the beam

Dataset	Method	Retrieval Metrics				Generation
		Strict Hit Rate (%)	Recall (%)	Precision (%)	Support F1 (%)	LLM Judge Acc (%)
MuSiQue	Naive RAG	14.23	43.96	21.17	28.11	21.51
	Hybrid RAG	14.23	43.98	21.18	28.13	21.56
	GraphRAG (Local)	30.70	59.62	7.19	12.59	40.67
	LightRAG (Max)	33.54	60.44	20.23	18.58	40.81
	KG^2RAG	9.10	31.02	11.50	15.61	15.06
	TGS-RAG (Ours)	**34.84**	**62.01**	**24.85**	**20.60**	**41.37**
HotpotQA	Naive RAG	45.01	62.48	26.99	38.56	59.41
	Hybrid RAG	46.05	65.45	27.62	39.42	61.92
	GraphRAG (Local)	55.78	70.15	10.11	20.85	71.79
	LightRAG (Max)	60.55	72.04	10.23	15.58	72.09
	KG^2RAG	38.83	58.92	16.49	25.40	60.46
	TGS-RAG	**62.00**	**77.55**	**27.41**	**26.06**	**79.99**

Table 1: Main results on MuSiQue and HotpotQA datasets. Retrieval performance is evaluated using Provenance Mapping. Generation quality is evaluated by DeepSeek-V3.2 as a Judge.

selection. If found, the path leading to this orphan entity is "resurrected" as a bridge path p_{bridge}. This mechanism efficiently recovers logical connections that are semantically distant but contextually bridged by the text, without incurring additional database query overhead.

3.4 Context Consolidation and Answer Generation

After precise scoring and ranking, the Top-K paths P_{Top-K} and text chunks C_{Top-K} are selected. Each P_{Top-K} is formatted into clear natural language reasoning chains. If supported by p_{bridge}, these bridging paths are also appended as supporting metadata. Finally, the consolidated context of P_{Top-K} and C_{Top-K}, along with the original query q, is submitted to the LLM to generate a logically coherent and evidence-grounded answer.

4 Experiments

In this section, we present a comprehensive evaluation of TGS-RAG against state-of-the-art retrieval paradigms. We aim to validate the effectiveness of our bidirectional synergy mechanism in resolving the "information island" problem inherent in multi-hop reasoning tasks.

4.1 Experimental Setup

Datasets

We utilize two widely recognized benchmarks for multi-hop question answering:

- **MuSiQue-Ans** [Trivedi *et al.*, 2022]: A dataset characterized by connected reasoning chains (typically 2-4 hops) and low lexical overlap between connected documents. It serves as the primary testbed for our "Orphan Entity Bridging" mechanism.
- **HotpotQA (Distractor)** [Yang *et al.*, 2018]: A dataset requiring reasoning over two supporting documents amidst logically irrelevant but semantically similar distractor documents.

Implementation Details

To ensure a robust and cost-effective experimental environment, we employ **GPT-4o-mini** as the backbone LLM for all tasks, including Knowledge Graph construction, entity extraction, and final answer generation. For semantic representation, we utilize **Qwen3-Embedding-0.6B**[Zhang *et al.*, 2025] as the unified embedding model for all queries, text chunks, and graph entities. For evaluation, we utilize DeepSeek-V3.2 [DeepSeek-AI, 2024] as an impartial LLM-as-a-Judge [Li *et al.*, 2025]. We instruct DeepSeek-V3.2 to evaluate semantic equivalence between the generated answer and the ground truth, accounting for valid aliases while penalizing hallucinations.

Baselines

We compare TGS-RAG with three categories of retrieval methods. All baselines are implemented following their original configurations, utilizing identical LLM backbones (GPT-4o-mini) and consistent text chunking strategies to ensure a fair comparison.

- **Naive RAG:** Standard dense retrieval using vector similarity.
- **Hybrid RAG (Dense+Sparse Text Retrieval):** A combination of dense vector retrieval and sparse keyword retrieval (BM25/TF-IDF) fused via Reciprocal Rank Fusion (RRF) [Cormack *et al.*, 2009].
- **Graph-centric Methods:** We evaluate state-of-the-art graph retrieval systems, including **GraphRAG** [Edge *et al.*, 2025] and **LightRAG** [Guo *et al.*, 2025]. To rigorously benchmark the performance ceiling of each method, we employ their most comprehensive retrieval modes and include their associated textual evidence or community summaries where applicable: "Local Search" for GraphRAG (focusing on entity-centric reasoning) and "Max" mode for LightRAG (integrating both global and local signals). Additionally, we compare against KG^2RAG [Zhu *et al.*, 2025] as a representative of semantic-to-graph expansion paradigms.

Dataset	Method	Token Usage			
		Embedding	LLM Prompt	LLM Completion	Total Cost
MuSiQue	Naive RAG	54,735	1,756,984	13,461	1,825,180
	Hybrid RAG	54,735	1,758,189	13,437	1,826,361
	GraphRAG (Local)	51,845	13,908,681	554,681	14,515,224
	LightRAG (Max)	174,024	16,541,984	374,635	17,090,643
	KG^2RAG	1,795,495	909,001	10,213	2,714,709
	TGS-RAG	**78,030**	**5,113,149**	**252,224**	**5,443,403**
HotpotQA	Naive RAG	2,202,470	59,485,904	345,673	62,034,047
	Hybrid RAG	2,280,240	63,283,087	346,239	65,909,566
	GraphRAG (Local)	1,475,556	634,239,236	10,674,261	646,389,053
	LightRAG (Max)	4,574,224	746,163,808	7,163,934	757,901,966
	KG^2RAG	2,119,623	41,072,699	398,399	43,590,721
	TGS-RAG	**3,406,165**	**187,692,537**	**26,438,352**	**217,537,053**

Table 2: Efficiency and token usage analysis. Detailed breakdown of token consumption across MuSiQue and HotpotQA datasets. Grouping by dataset allows for a direct comparison of computational costs among different retrieval paradigms.

For methods with explicit text chunk retrieval, we fix the text chunk top-k to $k = 5$.

Evaluation Metrics

To rigorously assess both retrieval completeness and generation accuracy, we employ a multi-dimensional metric system.

Retrieval Scope with Provenance Mapping. Unlike standard RAG which only retrieves text chunks, TGS-RAG retrieves both structured paths and unstructured chunks. To ensure a fair comparison with document-level ground truth, we define the set of retrieved documents D_{ret} as the union of explicit document retrieval and implicit graph provenance:

$$D_{ret} = D_{chunks} \cup \{doc \mid \exists e \in P_{graph}, doc \in Source(e)\} \quad (4)$$

where D_{chunks} denotes the source documents of the top-k retrieved text chunks, and $Source(e)$ represents the source documents of entities contained in the retrieved graph paths P_{graph}. Based on D_{ret}, we calculate **Strict Hit Rate (SHR)**, the percentage of queries where D_{ret} contains *all* ground truth supporting documents, and **Support F1**, measuring the quality of evidence. Because graph-based retrieval expands evidence through entity-to-document provenance, its retrieved document set is typically broader than chunk-only baselines, making Support F1 a conservative metric that may be lower despite stronger structural coverage.

Generation Quality (LLM-as-a-Judge). Traditional string-matching metrics (e.g., Exact Match) often penalize correct but verbose answers. Therefore, we adopt **LLM Judge Accuracy**, employing DeepSeek-V3.2 to judge whether the generated response contains the correct information specified in the gold answer.

4.2 Main Results

Performance Comparison

Table 1 presents a comprehensive evaluation of retrieval and generation performance across MuSiQue and HotpotQA benchmarks.

Limitations of Shallow Hybridization and Unidirectional Expansion. The results reveal critical bottlenecks in existing retrieval paradigms. As observed in the MuSiQue dataset, **Hybrid RAG** yields a Strict Hit Rate (14.23%) and Judge Accuracy (21.56%) virtually identical to Naive RAG. This stagnation confirms that the primary challenge in multi-hop reasoning is not lexical mismatch—which keyword search addresses—but *structural disconnection*. Hybrid methods fail to bridge the gap when documents share no lexical overlap but are logically linked. Similarly, KG^2RAG underperforms significantly (9.10% Hit Rate on MuSiQue), demonstrating the fragility of unidirectional "semantic-to-graph" expansion; if the initial semantic retrieval misses key anchor points, the subsequent graph reasoning collapses due to the lack of a feedback loop.

The Precision-Recall Trade-off in Graph Baselines. While heavy graph-based approaches like **GraphRAG** and **LightRAG** achieve competitive recall, they suffer from severe precision degradation. On HotpotQA, both methods exhibit extremely low retrieval precision (≈10%), compared to 27.41% for TGS-RAG. This indicates an *information overload* phenomenon: these systems retrieve excessive, logically loosely related graph components (e.g., entire communities), which introduces significant noise into the context window. Consequently, despite high recall, their generation accuracy (LLM Judge) is compromised by the distraction of irrelevant evidence.

Superiority of TGS-RAG. **TGS-RAG** establishes a new state-of-the-art by effectively balancing retrieval scope and precision. On MuSiQue, it achieves a Strict Hit Rate of **34.84%**, more than doubling the performance of Hybrid RAG. On HotpotQA, it attains the highest Strict Hit Rate, Recall, and Judge Accuracy, reaching a Judge Accuracy of **79.99%** despite a conservative Support F1 of **26.06%** under broader graph provenance. Unlike baselines that trade precision for recall, TGS-RAG utilizes the bidirectional synergy to filter out pseudo-evidence while resurrecting subtle logical links, providing the LLM with a concise and rigorous reason-

Variant	SHR (%)	Recall (%)	Prec (%)	F1 (%)
TGS-RAG (Full)	**62.00**	**77.55**	**27.41**	**26.06**
w/o Bridging	47.82	68.16	22.74	23.46
w/o Re-ranking	52.65	70.31	23.88	24.08

Table 3: Ablation study on HotpotQA. Detailed impact of removing synergy modules.

ing chain.

4.3 Efficiency Analysis

Table 2 details the computational costs associated with each paradigm. A comparative analysis highlights the superior cost-effectiveness of our framework.

Prohibitive Costs of Global Indexing. Graph-based global indexing methods impose a prohibitive computational burden. **LightRAG** and **GraphRAG** incur astronomical token usage (e.g., over 757M tokens for LightRAG on HotpotQA vs. 62M for Naive RAG), primarily driven by the exhaustive construction of community summaries and dual-level indices. This massive overhead renders them impractical for dynamic or large-scale knowledge bases where frequent updates are required.

Optimal Pareto Frontier of TGS-RAG. TGS-RAG achieves an optimal trade-off between performance and efficiency. While it incurs a moderate cost increase over Naive RAG (≈3x) to support graph reasoning, it is drastically more efficient than global graph methods—consuming only **37%** of the tokens used by GraphRAG on MuSiQue and less than **30%** of LightRAG on HotpotQA. This efficiency stems from our algorithmic design: instead of pre-computing expensive global summaries, TGS-RAG employs an *on-demand* retrieval strategy. The *Semantic Beam Search* and *Memory-based Orphan Entity Bridging* selectively explore and resurrect only contextually relevant paths during inference, avoiding the redundant processing of the entire graph structure.

4.4 Ablation Study

To verify the contribution of each module in our bidirectional synergy mechanism, we conducted an ablation study on the **HotpotQA** dataset. We developed two variants:

- **w/o Graph-to-Text (Re-ranking):** Disables the *Global Voting* mechanism. Text chunks are ranked solely by vector similarity.
- **w/o Text-to-Graph (Bridging):** Disables the *Memory-based Orphan Entity Bridging*. The system relies solely on the initial graph paths found by beam search.

As shown in Table 3, removing the Graph-Guided Re-ranking module leads to a decline in Strict Hit Rate to **52.65%** and Support F1 to **24.08%**. This indicates that without structured filtering, the retriever struggles to distinguish between semantically similar distractors and true supporting facts, thereby reducing the precision and overall quality of the evidence context.

Similarly, removing the Bridging module results in the most significant performance degradation, with Strict Hit Rate dropping sharply from **62.00%** to **47.82%**. This confirms that standard graph traversal may fail when potentially valid evidence nodes are pruned early, whereas our memory-based resurrection mechanism effectively recovers these "orphan" connections.

5 Conclusion

We have proposed **TGS-RAG**, a bidirectional text–graph synergistic framework designed to resolve the isolation between unstructured textual evidence and structured graph-based reasoning. By modeling retrieval as a collaborative process, TGS-RAG effectively bridges the "Information Island" gap inherent in multi-hop reasoning tasks.

The framework introduces two complementary enhancement channels: **Graph-Guided Re-ranking** leverages structured logic to filter semantic noise from text retrieval, while **Memory-based Orphan Entity Bridging** utilizes textual context to validate and complete graph reasoning paths. A critical innovation is the use of "visited memory" to *resurrect* pruned paths, which recovers potentially useful logical links that were previously discarded during search, without the computational overhead of iterative expansion.

Extensive experiments on MuSiQue and HotpotQA demonstrate that TGS-RAG consistently outperforms strong text-based, graph-based, and hybrid baselines. Our results indicate that shallow hybridization is insufficient; instead, effective multi-hop reasoning requires structured and unstructured evidence to mutually verify and correct each other. Notably, TGS-RAG achieves this state-of-the-art performance with significantly lower token consumption compared to global graph retrieval methods. Overall, this work underscores the importance of bidirectional synergy and provides a scalable foundation for integrating symbolic structure with neural representations in LLMs.

References

[Cormack *et al.*, 2009] Gordon V. Cormack, Charles L A Clarke, and Stefan Buettcher. Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '09, pages 758–759, New York, NY, USA, 2009. Association for Computing Machinery.

[Cossio, 2025] Manuel Cossio. A comprehensive taxonomy of hallucinations in large language models, 2025.

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

[Edge *et al.*, 2025] Darren Edge, Ha Trinh, Newman Cheng, Joshua Bradley, Alex Chao, Apurva Mody, Steven Truitt, Dasha Metropolitansky, Robert Osazuwa Ness, and Jonathan Larson. From local to global: A graph rag approach to query-focused summarization, 2025.

[Guo *et al.*, 2025] Zirui Guo, Lianghao Xia, Yanhua Yu, Tu Ao, and Chao Huang. LightRAG: Simple and fast retrieval-augmented generation. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose,

and Violet Peng, editors, *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 10746–10761, Suzhou, China, November 2025. Association for Computational Linguistics.

[He *et al.*, 2024] Xiaoxin He, Yijun Tian, Yifei Sun, Nitesh V. Chawla, Thomas Laurent, Yann LeCun, Xavier Bresson, and Bryan Hooi. G-retriever: Retrieval-augmented generation for textual graph understanding and question answering. In A. Globerson, L. Mackey, D. Belgrave, A. Fan, U. Paquet, J. Tomczak, and C. Zhang, editors, *Advances in Neural Information Processing Systems*, volume 37, pages 132876–132907. Curran Associates, Inc., 2024.

[Ji *et al.*, 2023] Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. Survey of hallucination in natural language generation. *ACM Comput. Surv.*, 55(12), March 2023.

[Karpukhin *et al.*, 2020] Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, editors, *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online, November 2020. Association for Computational Linguistics.

[Khattab and Zaharia, 2020] Omar Khattab and Matei Zaharia. Colbert: Efficient and effective passage search via contextualized late interaction over bert. In *Proceedings of the 43rd International ACM SIGIR conference on research and development in Information Retrieval*, pages 39–48, 2020.

[Lewis *et al.*, 2020] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive nlp tasks. In H. Larochelle, M. Ranzato, R. Hadsell, M.F. Balcan, and H. Lin, editors, *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474. Curran Associates, Inc., 2020.

[Li *et al.*, 2025] Dawei Li, Bohan Jiang, Liangjie Huang, Alimohammad Beigi, Chengshuai Zhao, Zhen Tan, Amrita Bhattacharjee, Yuxuan Jiang, Canyu Chen, Tianhao Wu, Kai Shu, Lu Cheng, and Huan Liu. From generation to judgment: Opportunities and challenges of LLM-as-a-judge. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng, editors, *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 2757–2791, Suzhou, China, November 2025. Association for Computational Linguistics.

[Ma *et al.*, 2025] Shengjie Ma, Chengjin Xu, Xuhui Jiang, Muzhi Li, Huaren Qu, Cehao Yang, Jiaxin Mao, and Jian Guo. Think-on-graph 2.0: Deep and faithful large language model reasoning with knowledge-guided retrieval augmented generation. In Y. Yue, A. Garg, N. Peng, F. Sha, and R. Yu, editors, *International Conference on Representation Learning*, volume 2025, pages 52782–52806, 2025.

[Sun *et al.*, 2024] Jiashuo Sun, Chengjin Xu, Lumingyuan Tang, Saizhuo Wang, Chen Lin, Yeyun Gong, Lionel Ni, Heung-Yeung Shum, and Jian Guo. Think-on-graph: Deep and responsible reasoning of large language model on knowledge graph. In B. Kim, Y. Yue, S. Chaudhuri, K. Fragkiadaki, M. Khan, and Y. Sun, editors, *International Conference on Learning Representations*, volume 2024, pages 3868–3898, 2024.

[Trivedi *et al.*, 2022] Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. MuSiQue: Multihop questions via single-hop question composition. *Transactions of the Association for Computational Linguistics*, 10:539–554, 2022.

[Yang *et al.*, 2018] Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D. Manning. HotpotQA: A dataset for diverse, explainable multi-hop question answering. In Ellen Riloff, David Chiang, Julia Hockenmaier, and Jun'ichi Tsujii, editors, *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 2369–2380, Brussels, Belgium, October-November 2018. Association for Computational Linguistics.

[Zhang *et al.*, 2025] Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, Fei Huang, and Jingren Zhou. Qwen3 embedding: Advancing text embedding and reranking through foundation models, 2025.

[Zhu *et al.*, 2025] Xiangrong Zhu, Yuexiang Xie, Yi Liu, Yaliang Li, and Wei Hu. Knowledge graph-guided retrieval augmented generation. In Luis Chiruzzo, Alan Ritter, and Lu Wang, editors, *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, Albuquerque, New Mexico, April 2025. Association for Computational Linguistics.