COSMOS: Connectivity-Oriented Submodular Maximization for Optimal Subgraph Retrieval

Boci Peng[1,2] **Xiao Liu**[3] **Boren Hu**[4] **Yun Zhu**[5]
Xuanbo Fan[1,2] **Yanwei Yue**[1,2] **Chunyu Yang**[6] **Yan Zhang**[1,2]*
[1]School of Intelligence Science and Technology, Peking University
[2]State Key Laboratory of General Artificial Intelligence, Peking University, Beijing, China
[3]Computer Science, University of Macau, [4]Independent Researcher
[5]Shanghai Artificial Intelligence Laboratory, [6]Ucap Cloud
bcpeng@stu.pku.edu.cn, zhyzhy001@pku.edu.cn

Abstract

Retrieving coherent evidence subgraphs is critical for Knowledge Base Question Answering (KBQA). Existing paradigms often treat facts independently, rely on biased heuristics, or employ myopic search, failing to optimize collective subgraph utility. In this paper, we propose **COSMOS** (**C**onnectivity-**O**riented **S**ubmodular **M**aximization for **O**ptimal **S**ubgraph Retrieval), a unified framework that formalizes evidence retrieval as a constrained submodular maximization problem. This formulation mathematically captures the trade-off between information relevance and structural complexity. To tractably solve this combinatorial challenge, COSMOS employs a decompose-and-conquer strategy, which first performs a seed-guided greedy expansion to maximize local semantic utility, followed by a topology-aware component aggregation to bridge disjoint evidence clusters via Maximum Spanning Tree aggregation. Guided by theoretical bounds, we introduce Structure-Aware Contrastive Tuning to align semantic space with KG topology. Experimental results on WebQSP, CWQ, and M^{3}GQA benchmarks demonstrate that COSMOS achieves state-of-the-art performance.

1 Introduction

Knowledge Base Question Answering (KBQA) (Lan et al., 2021, 2023; Peng et al., 2025b) aims to answer natural language questions by reasoning over structured facts in Knowledge Graphs (KGs). While Retrieval-Augmented Generation (RAG) (Fan et al., 2024) has emerged as a promising paradigm to mitigate the hallucinations of Large Language Models (LLMs) (Zhao et al., 2024b), applying RAG to KGs presents unique challenges. Unlike unstructured text retrieval, where documents are treated independently, reasoning over KGs relies heavily on the topological structure of facts. Effective reasoning requires retrieving not just a disjoint set of relevant triplets, but a coherent, connected evidence subgraph that delineates the logical path from the query entities to the answer.

Paradigm	Formal Objective	Structural Integrity	Reasoning Fidelity	Global Perspective	Inference Efficiency
Embedding	$\text{argmax}_e \text{sim}(\mathbf{q}, \mathbf{e})$	✗	✗	✓	✓
Heuristic	$\text{argmin}_T \sum_{e \in T} w_e$	✓	✗	✓	✓
Search	$\prod_t P(a_t \mid \mathcal{G}, h_{<t}, q)$	✓	✓	✗	✗
COSMOS (ours)	$\text{argmax}_S F(S)$ s.t. $\lvert E_S \rvert \leq K$, S is connected	✓	✓	✓	✓

Table 1: Comparison of subgraph retrieval paradigms. *Structural Integrity* ensures the retrieval of connected evidence chains; *Reasoning Fidelity* denotes the ability to capture deep semantic logic beyond surface similarity or topological bias; *Global Perspective* indicates the optimization of collective subgraph utility to avoid the myopia of greedy search; and *Inference Efficiency* reflects low-latency retrieval without iterative LLM overhead.

Extensive research has sought to bridge natural language queries and structural knowledge, yet existing paradigms struggle to balance semantic relevance with structural integrity as query complexity grows. Embedding-based methods (Li et al., 2025; Jiang et al., 2023b; Baek et al., 2023a) retrieve facts independently, suffering from structural agnosticism and redundancy that impede coherent reasoning chains. Conversely, heuristic-based algorithms (He et al., 2024; Luo et al., 2024b; Jiang et al., 2025b) ensure connectivity but exhibit topological bias, often favoring short, generic paths over complex multi-hop reasoning. Search-based methods (Sun et al., 2024; Jiang et al., 2025a; Zhang et al., 2022) employ iterative LLM navigation but remain fundamentally myopic and inefficient, prone to local optima and prohibitive latency. Ultimately, these approaches prioritize local relevance over structural density and global connectivity, failing to optimize collective subgraph utility under constrained budgets.

*Corresponding author

Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), pages 35904–35924
July 2-7, 2026

To address these challenges, we propose COSMOS, a unified framework that formalizes evidence subgraph retrieval as a Connectivity-Oriented Submodular Maximization problem. This marks a paradigm shift from the passive observation of graph topology to the active synthesis of a global evidence backbone. By modeling subgraph utility through a submodular lens, which incorporates a strict budget constraint to implicitly manage information redundancy, COSMOS mathematically captures the notions of coverage and global connectivity. This formulation provides a rigorous theoretical foundation for what constitutes an optimal evidence subgraph, moving beyond purely heuristic or semantic criteria. Critically, we prove that this combinatorial objective can be solved with a guaranteed approximation ratio, ensuring that the results are not only efficient but also theoretically grounded against the NP-hard optimal solution.

To realize this active synthesis tractably, COSMOS employs a decompose-and-conquer optimization strategy. The framework first performs Seed-Guided Greedy Expansion to identify high-value local evidence clusters centered around source entities. To resolve the global connectivity gap, we transform the challenge into a Maximum Spanning Tree (MaxST) problem over a semantic meta-graph. This Topology-Aware Component Aggregation bridges isolated components using the most semantically salient paths, ensuring a globally coherent structure while maintaining polynomial-time efficiency. Furthermore, we introduce Structure-Aware Contrastive Tuning to align the embedding space with KG structures, ensuring the encoder can discern the fine-grained logical backbone required for multi-hop reasoning. Extensive experiments on WebQSP, CWQ, and M^3GQA demonstrate that COSMOS attains state-of-the-art results under the standard evaluation protocol.

Our contributions are summarized as follows:

- We provide a rigorous formulation of KBQA subgraph retrieval as a constrained submodular maximization problem, offering a theoretical grounding for balancing relevance and connectivity.
- We propose COSMOS, an efficient subgraph retrieval framework that addresses the structural fragmentation of embedding-based retrievers, the topological bias of heuristic algorithms, and the myopia of search-based agents through a decompose-and-conquer optimization strategy, backed by provable approximation guarantees.
- Extensive experiments on WebQSP, CWQ, and M^3GQA datasets demonstrate that COSMOS achieves state-of-the-art performance, retrieving more effective evidence subgraphs compared to existing baselines.

2 Related Works

We categorize existing KBQA methods into three primary paradigms based on their retrieval strategies, including embedding-based, heuristic-based, and search-based.

Embedding-Based Methods. Embedding-based methods map natural language queries and KG elements into a shared continuous vector space. The mainstream methods focus on optimizing embedding models to compute semantic similarity between triplets and queries to select the top-k relations or facts (Oguz et al., 2022; Jiang et al., 2023b; Baek et al., 2023a; Yu et al., 2023; Saxena et al., 2020; Bordes et al., 2015; Baek et al., 2023b; Dong et al., 2023). To capture deeper structural dependencies, some methods (Mavromatis and Karypis, 2025; Sun et al., 2019; Li et al., 2025; Peng et al., 2024; Liu et al., 2024) utilize Graph Neural Networks (GNNs) to enrich entity and relation representations via neighborhood message passing (Kipf and Welling, 2017; Veličković et al., 2018). Despite their efficiency, these methods treat triplets as independent units. This structural agnosticism often results in fragmented, disjoint evidence that fails to form a coherent reasoning chain.

Heuristic-Based Methods. Heuristic-based methods utilize graph algorithms or predefined rules to retrieve subgraphs. Some frameworks model retrieval as graph optimization problems, such as G-Retriever (He et al., 2024) (PCST problem) or RoK (Jiang et al., 2024) and RASR (Luo et al., 2023) (PageRank). Another trend involves using LLMs to generate relational paths that are subsequently grounded or ranked in the KG (Luo et al., 2024b; Shen et al., 2025; Luo et al., 2024a; Guo et al., 2024; Kim et al., 2023; Wu et al., 2023). Additionally, simpler heuristics like 1-hop neighborhood extraction (Yang et al., 2024) or predefined path patterns (Jiang et al., 2025b) are common. These methods are often topology-biased. By optimizing for topological

objectives (e.g., minimizing path cost) or following rigid rules, they frequently favor shorter, generic paths over the complex, multi-hop reasoning chains required for deep semantic understanding.

Search-Based Methods. Search-based methods treat KBQA as a sequential decision-making process. Agent-based frameworks (Jin et al., 2024; Wang et al., 2023a; Jiang et al., 2023a, 2025a; Wang et al., 2023b) utilize LLMs to interact with KGs as environments. Similarly, (Sun et al., 2024; Zhang et al., 2022; He et al., 2021; Dehghan et al., 2024; Xu et al., 2025), drive the search process using LLMs to navigate entity-relation paths dynamically. These methods are fundamentally myopic and inefficient. Their step-by-step greedy choices often lead to local optima and prohibitive inference latency due to iterative LLM calls. While recent work has attempted to address this limitation by framing reasoning as structure-aware planning over internally generated entailment graphs (Xiong et al., 2025), they still rely on implicit world models rather than explicit, verifiable evidence from external knowledge graphs. Critically, they fail to optimize the global connectivity and collective utility of the underlying evidence subgraph, which is essential for faithful multi-hop inference over structured knowledge.

Unlike previous subgraph retrievers, we formalize the process as a general connectivity-constrained submodular maximization problem, offering a unifying optimization view.

3 Preliminaries

In this section, we formally define the task of KBQA via subgraph retrieval and formulate it as a connectivity-constrained submodular maximization problem.

3.1 Task Definition

Knowledge Graph. Let $\mathcal{G} = (\mathcal{V}, \mathcal{E}, \mathcal{R})$ be a Knowledge Graph, where $\mathcal{V}$ denotes the set of entities, $\mathcal{R}$ denotes the set of relations, and $\mathcal{E} \subseteq \mathcal{V} \times \mathcal{R} \times \mathcal{V}$ denotes the set of facts (triplets).

Subgraph Retrieval. Given a natural language query Q, the goal of subgraph-based KBQA is to retrieve a concise evidence subgraph $S = (V_S, E_S)$ where $V_S \subseteq \mathcal{V}$ and $E_S \subseteq \mathcal{E}$. This subgraph should contain the necessary reasoning paths to derive the correct answer a for Q. We assume a set of source entities $V_Q \subset \mathcal{V}$ is identified from Q to ground the retrieval process.

3.2 Problem Formulation

We mathematically formulate the retrieval of the evidence subgraph S as selecting a subset of edges to maximize a semantic utility function $F(S)$ under structural constraints.

Submodularity. To model the balance between relevance and redundancy, we leverage the property of *submodularity* (Nemhauser et al., 1978). Let Ω be a finite ground set (in our case, the edge set $\mathcal{E}$). A set function $F : 2^{\Omega} \to \mathbb{R}$ is **submodular** if it satisfies the property of diminishing returns: for every $A \subseteq B \subseteq \Omega$ and every element $x \in \Omega \setminus B$, the inequality $F(A \cup \{x\}) - F(A) \geq F(B \cup \{x\}) - F(B)$ holds.

Optimization Objective. Based on this, the core subgraph retrieval task is defined as the following constrained optimization problem:

$$\max_{S \subseteq \mathcal{G}} \quad F(S) \\ \text{s.t.} \quad |E_S| \leq K, \quad S \text{ is connected}, \quad V_Q \subseteq V_S, \tag{1}$$

where K is the budget for the subgraph size. The constraint ensures that the retrieved subgraph is concise, topologically connected, and rooted in the query entities.

Utility Function Instantiation. To instantiate our submodular framework, we adopt the monotone Maximal Marginal Relevance (MMR) function. Formally, let $S = \{e_1, e_2, \ldots, e_k\}$ be an ordered set of retrieved edges. We define the collective utility $F(S)$ as the sum of non-negative marginal gains: $F(S) = \sum_{t=1}^{|S|} \max(0, \Delta F(e_t \mid S_{t-1}))$, where $S_{t-1} = \{e_1, \ldots, e_{t-1}\}$ and $S_0 = \emptyset$. The marginal gain ΔF for a candidate edge e is defined as:

$$\Delta F(e \mid S_{t-1}) = \lambda \cdot \text{sim}(\mathbf{e}, \mathbf{q}) - (1 - \lambda) \cdot \max_{e' \in S_{t-1}} \text{sim}(\mathbf{e}, \mathbf{e}'), \tag{2}$$

where $\mathbf{e}$ is the dense vector representation of a knowledge graph triplet, $\mathbf{q}$ is the dense vector representation of the input query, and $\text{sim}(\cdot)$ denotes cosine similarity.

4 Methodology

4.1 Overview

Our objective is to retrieve an evidence subgraph S^* that maximizes the semantics. We propose COSMOS to solve the connectivity-constrained

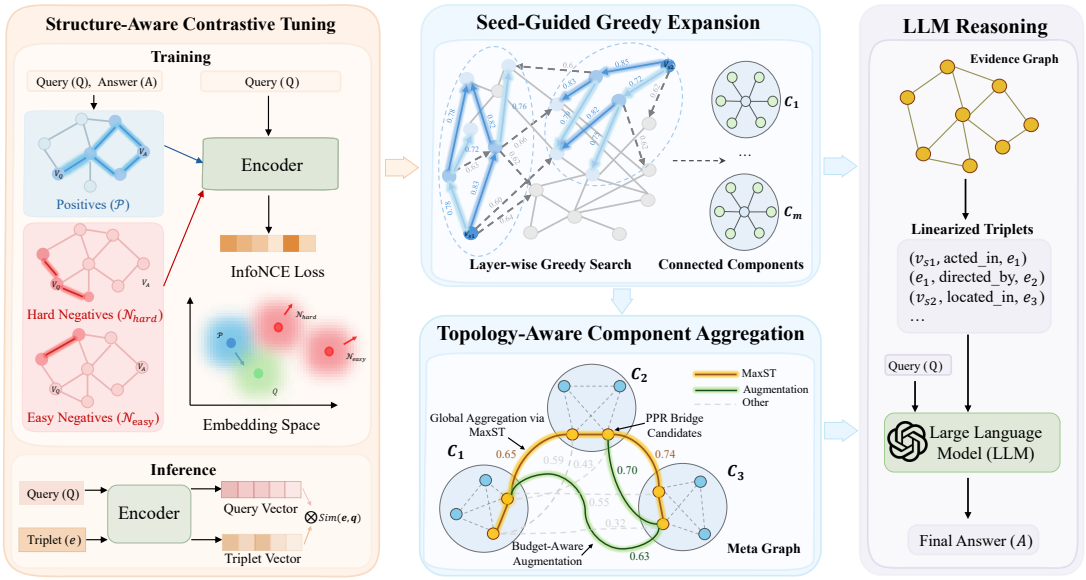

Figure 1: The overview of COSMOS. The query and KG are first encoded via SACT. Then, SGE extracts local clusters around seed entities, which are subsequently bridged by TACA using MaxST on the meta-graph to form the final evidence subgraph.

submodular maximization problem by decoupling the optimization into two tractable stages: (1) *Seed-Guided Greedy Expansion* to maximize local relevance around source entities, and (2) *Topology-Aware Component Aggregation* to resolve global connectivity between disjoint clusters. To align the semantic space with the discrete KG topology, we further incorporate *Structure-Aware Contrastive Tuning*. Figure 1 illustrates the framework.

4.2 Semantic Encoding

Prior to retrieval, we project both the query and KG facts into a shared dense vector space. Let a fact triplet be denoted as $e = (h, r, t)$. We linearize the triplet into a textual sequence "(h, r, t)" and employ a pre-trained language model as the encoder. For a query Q and a triplet e, their embeddings are derived from the `[CLS]` token representation:

$$\mathbf{q} = \text{Enc}(Q), \quad \mathbf{e} = \text{Enc}(h, r, t). \tag{3}$$

Based on these representations, we quantify the semantic similarity using cosine similarity.

4.3 Seed-Guided Greedy Expansion

Seed-Guided Greedy Expansion (SGE) aims to address the rooted submodular maximization problem. We aim to mine high-value local evidence surrounding the identified source entities V_Q without immediately enforcing global connectivity.

Layer-wise Greedy Search. Specifically, we treat each source entity v_s as a root and perform a layer-wise greedy search to construct a local subgraph S_i for each v_s. Let B_{layer} denote the maximum edge budget per hop. At each expansion step, we examine the candidate neighbor edges $\mathcal{N}(S_i)$ of the current subgraph. We select the edge e^* that provides the maximum marginal gain. This process iterates until the layer budget B_{layer} or the maximum hop limit is reached. The output of this phase is a set of disjoint connected components, denoted as $\mathcal{C} = \{C_1, C_2, \ldots, C_m\}$, where each component represents a highly relevant local cluster centered around a source entity.

Theoretical Analysis. We analyze its approximation ratio. While topological constraints limit standard greedy algorithms, we introduce a Topology-Semantic Alignment Factor $\gamma \in (0, 1]$ to quantify the consistency between local marginal gains and global optimal gains within the graph structure.

Theorem 1. *Let S^* be the optimal subgraph. Under the alignment assumption γ, the SGE phase achieves a parameterized approximation ratio:*

$$F(S_{SGE}) \geq (1 - e^{-\gamma}) F(S^*_{exp}). \tag{4}$$

γ quantifies the alignment between the semantic landscape and graph topology. Appendix A.1 illustrates the detailed definition and proof.

4.4 Topology-Aware Component Aggregation

Topology-Aware Component Aggregation (TACA) is designed to solve the connectivity-constrained optimization. The components in $\mathcal{C}$ are initially disconnected, potentially fragmenting the reasoning chain. Our goal is to select optimal paths to bridge these components into a unified evidence graph.

Meta-Graph Construction. To bridge the topological gap between the disjoint local clusters, we construct a high-level semantic roadmap that captures the connectivity potential of the KG. Specifically, we formalize this as a **Meta-Graph** $\mathcal{M} = (\mathcal{V}_{\mathcal{M}}, \mathcal{E}_{\mathcal{M}})$, where each meta-node $V_i \in \mathcal{V}_{\mathcal{M}}$ represents a connected component $C_i \in \mathcal{C}$. The objective is to identify candidate reasoning paths that can transform these isolated components into a globally coherent evidence structure.

For every pair of components (C_i, C_j), we perform a restricted path search in the original KG $\mathcal{G}$ to find all valid bridging paths $\mathcal{P}_{ij} = \{p \mid \text{link}(C_i, C_j), \text{length}(p) \leq L_{hop}\}$. To maintain computational tractability while ensuring semantic relevance, this search is constrained by a maximum hop limit L_{hop} and, as detailed in the following section, is further pruned via representative nodes.

To facilitate the subsequent connectivity optimization, we assign each potential meta-edge E_{ij} a primary weight W_{ij}. Specifically, W_{ij} is defined as the cumulative marginal utility gain of the bridging path $p \in \mathcal{P}_{ij}$ relative to the current components, computed as the sum of the marginal gains of all triplets along p. This ensures that the meta-edge weights reflect the total additional semantic information provided by the bridge.

We denote the path achieving this maximum utility as the optimal bridge p^*_{ij}. While W_{ij} is utilized as the primary criterion for constructing the minimal connected backbone, the entire collection of discovered paths $\bigcup_{i,j} \mathcal{P}_{ij}$ is preserved as a candidate path pool, which serves as the foundation for the subsequent budget-aware augmentation, allowing the framework to incorporate multifaceted evidence that exceeds the minimal tree structure.

Global Aggregation via MaxST. To resolve the connectivity constraint while maximizing total utility, we adopt a two-stage optimization strategy. First, we establish a minimal connected backbone inspired by the Maximum Spanning Tree (MaxST) problem. Specifically, we apply Kruskal's algorithm on the meta-graph $\mathcal{M}$ to identify a MaxST $\mathcal{T} \subseteq \mathcal{E}_{\mathcal{M}}$, which ensures that all disjoint components $\mathcal{C}$ are interconnected using paths that provide the highest cumulative semantic relevance. The resulting backbone subgraph S_{backbone} is formed by the union of the original components and the bridging paths corresponding to the edges in $\mathcal{T}$:

$$S_{\text{backbone}} = \left(\bigcup_{C_k \in \mathcal{C}} C_k \right) \cup \left(\bigcup_{(i,j) \in \mathcal{T}} p^*_{ij} \right), \quad (5)$$

where p^*_{ij} is the optimal physical path associated with the meta-edge E_{ij}.

After establishing the backbone S_{backbone} via MaxST, we further utilize the remaining budget K to incorporate supplementary evidence. Instead of being limited to the edges in the meta-tree, we consider all previously discovered paths in the candidate pool that are not yet included: $\mathcal{P}_{\text{aug}} = (\bigcup_{i,j} \mathcal{P}_{ij}) \setminus \{p^*_{ij} \mid (i,j) \in \mathcal{T}\}$. We rank all candidate paths $p \in \mathcal{P}_{aug}$ based on their marginal contribution to the submodular objective, $\Delta F(p|S_{backbone})$, and iteratively add them to $S_{backbone}$, forming the final evidence subgraph S^*. This process follows the greedy selection rule for submodular maximization, where the paths providing the highest marginal gain are prioritized until the budget K is exhausted. This allows the subgraph to contain multiple reasoning paths between the same pair of components, providing a more robust and diverse evidence set for LLM reasoning.

Efficiency Optimization via Semantic PPR. To avoid prohibitive quadratic complexity in exhaustive inter-component pathfinding, we employ Semantic Personalized PageRank (PPR) to identify representative bridge candidates within each component. Unlike threshold-based pruning, Semantic PPR propagates scores through the graph topology, allowing bridge nodes with low direct query similarity to accumulate high scores if they act as structural hubs. This ensures COSMOS retains structurally vital intermediate concepts often overlooked by strictly local semantic searches.

Formally, let $V(C_i)$ be the node set of component C_i, and $V_{root} \subseteq V(C_i)$ be the subset of source entities contained within it. We define an importance score vector $\mathbf{r} \in \mathbb{R}^{|V(C_i)|}$. First, We initialize the personalization vector $\mathbf{p}_0$ to bias the random walk toward the $k = |V_{root}|$ seed entities, setting $\mathbf{p}_0(v) = 1/k$ if $v \in V_{root}$ and 0 otherwise.

Unlike standard PPR, which uses uniform transition probabilities based on node degrees, we construct a semantic transition matrix $\mathbf{M}$ to guide

the random walk towards query-relevant reasoning paths. For any edge $e = (u, r, v)$ connecting node u to v, the transition probability is proportional to the semantic similarity between the triplet and the query:

$$\mathbf{M}_{uv} = \frac{\exp(\text{sim}(\mathbf{e}, \mathbf{q})/\tau)}{\sum_{v' \in \mathcal{N}(u)} \exp(\text{sim}(\mathbf{e}_{uv'}, \mathbf{q})/\tau)}, \quad (6)$$

where $\mathcal{N}(u)$ denotes the neighbors of u, and τ is a temperature factor. We then iteratively update the importance scores until convergence:

$$\mathbf{r}^{(t+1)} = (1 - \alpha)\mathbf{p}_0 + \alpha \mathbf{M}^\top \mathbf{r}^{(t)}, \quad (7)$$

where α is the damping factor. Upon convergence, we select the top-N nodes with the highest scores as the representative set R_i for pathfinding.

Theoretical Analysis. We model TACA as a budgeted maximization problem over a constructed Meta-Graph. To account for the information loss during meta-graph construction and the greedy selection under budget constraints, we introduce a *Meta-Graph Fidelity Factor* $\beta \in (0, 1]$.

Theorem 2. *Let S^*_{agg} be the optimal connectivity backbone in the global KG. Under fidelity assumption β, the backbone S_{TACA} produced by COSMOS satisfies:*

$$F(S_{TACA}) \geq \beta \cdot \left(1 - \frac{1}{e}\right) \cdot F(S^*_{agg}). \quad (8)$$

The approximation ratio is composed of two parts: β bounds the structural gap between our constructed Meta-Graph and the global optimal topology (due to pruning strategies), while $(1 - 1/e)$ represents the lower bound of the greedy selection strategy under the budget constraint (detailed proof in Appendix A.2).

4.5 Structure-Aware Contrastive Tuning

Our theoretical analysis (Theorem 1) identifies the alignment factor γ as the decisive bound for the effectiveness of greedy expansion. Standard pretrained encoders often suffer from semantic drift, leading to a low γ and rendering the greedy assumption invalid. Therefore, we propose Structure-Aware Contrastive Tuning (SACT) to explicitly maximize γ. By aligning the semantic space with topological reachability, SACT satisfies the prerequisites for our retrieval algorithms to function near their optimal bounds.

Hard Negative Mining. Contrastive learning relies on informative negatives. We construct our training data using a topology-aware strategy:

- **Positives ($\mathcal{P}$)**: For a training pair, we identify the shortest paths between source entities V_Q and answer entities V_A. All triplets along these paths are treated as positive samples.
- **Hard Negatives ($\mathcal{N}_{\text{hard}}$)**: We select triplets that are neighbors of the source or answer entities but do not appear in shortest paths.
- **Easy Negatives ($\mathcal{N}_{\text{easy}}$)**: Randomly sampled triplets from the KG.

Loss Function. Given the potentially large number of candidate triplets, utilizing the full set of positives and negatives for every update is computationally prohibitive. Therefore, during training, we randomly sample a fixed-size subset of positive triplets $\hat{\mathcal{P}} \subseteq \mathcal{P}_q$ from the reasoning paths, and a subset of negative triplets $\hat{\mathcal{N}} \subseteq \mathcal{N}_{\text{hard}} \cup \mathcal{N}_{\text{easy}}$. We then employ a multi-positive extension of the InfoNCE loss over these sampled sets, inspired by (Frosst et al., 2019; Zhao et al., 2024a):

$$\mathcal{L} = -\log \frac{\sum_{p^+ \in \hat{\mathcal{P}}} \exp(\text{sim}(\mathbf{q}, \mathbf{p}^+)/\tau)}{\sum_{p \in \hat{\mathcal{P}} \cup \hat{\mathcal{N}}} \exp(\text{sim}(\mathbf{q}, \mathbf{p})/\tau)}. \quad (9)$$

By aggregating the likelihood over multiple sampled positives, this objective encourages the query representation to align simultaneously with various steps of the evidence chain, rather than a single fact. This structure-aware tuning effectively forces the retriever to discern the logical connectivity.

4.6 LLM Reasoning

Given the retrieved evidence subgraph S^*, we linearize it into a sequence of knowledge triplets to serve as the reasoning context. Specifically, for each edge in S^*, we flatten it into a textual triplet (h, r, t) and concatenate all triplets into a unified context string c. To effectively derive the answer from these structured facts, we employ a Chain-of-Thought (CoT) (Wei et al., 2022) strategy. Formally, the LLM takes the query Q and the serialized evidence c as input, generating a reasoning chain followed by the final answer A.

5 Experiments

5.1 Experimental Settings

Datasets. We evaluate COSMOS on two standard benchmarks grounded in Freebase (Bollacker

et al., 2008): WebQSP (Yih et al., 2016) and CWQ (Talmor and Berant, 2018). Additionally, to assess performance on highly complex, multi-entity reasoning, we conduct evaluations on the M^{3}GQA (Peng et al., 2025a), with results reported in Appendix D.4. Detailed descriptions and statistics for all datasets are provided in Appendix D.1.

Baselines. We compare COSMOS against open-sourced baselines across three paradigms: (1) Traditional methods, encompassing embedding-based and retrieval-augmented approaches that select facts or subgraphs for reasoning; (2) LLMs, which assess the inherent parametric knowledge of the models; and (3) LLMs+KGs methods, representing the state-of-the-art in unifying KG retrieval with LLM-based reasoning.

Metrics. Following standard protocols (Luo et al., 2024b; Li et al., 2025), we utilize Hit and F1 Score for evaluation. Specifically, we report Hits1 for traditional baselines and Hit for LLM-based methods (measuring if the generated response contains at least one correct answer), as the latter produce answer sets rather than ranked lists. F1 Score is further employed to assess the comprehensive coverage and precision of the predicted answers.

Implementations. We implement COSMOS using GPT-4o-mini as the reasoner and gte-large-en-v1.5 (Li et al., 2023) for encoding. Regarding baselines, we follow the unified evaluation setting of SubgraphRAG (Li et al., 2025) and RoG (Luo et al., 2024b), and thus directly adopt their reproduced baseline results. Further implementation details are provided in Appendix D.3.

Methods	WebQSP		CWQ	
	Hit	F1	Hit	F1
Traditional Methods				
KV-Mem	46.7	34.5	18.4	15.7
EmbededKGQA	66.6	-	45.9	-
NSM	68.7	62.8	47.6	42.4
TransferNet	71.4	-	48.6	-
KGT5	56.1	-	36.5	-
GraftNet	66.4	60.4	36.8	32.7
PullNet	68.1	-	45.9	-
SR+NSM	68.9	64.1	50.2	47.1
SR+NSM+E2E	69.5	64.1	49.3	46.3
LLMs Methods				
GPT-4o-mini	60.4	41.9	38.1	31.9
GPT-4o	65.7	49.2	42.3	38.3
GPT-4o-mini+CoT	61.1	41.6	39.7	34.7
GPT-4o+CoT	66.4	46.3	45.8	41.0
LLMs+KGs Methods				
KD-CoT	68.6	52.5	55.7	-
UniKGQA	77.2	72.2	51.2	49.1
Retrieve-Rewrite-Answer	79.4	-	-	-
G-Retriever	73.5	53.4	-	-
RoG	82.2	66.5	60.6	53.9
EtD	82.5	-	62.0	-
SubgraphRAG	90.1	77.5	62.0	54.1
COSMOS (ours)	**91.6**	**78.2**	**65.7**	**55.6**

Table 2: QA performance on WebQSP and CWQ (%).

5.2 Overall Performance

We compare COSMOS with baselines on the WebQSP and CWQ benchmarks, with the comparative results summarized in Table 2. We further validate our approach on the challenging M^{3}GQA benchmark, with full generation and retrieval results presented in Table 8 and Table 9 (Appendix D.4).

From these results, we derive several key observations: (1) Approaches that integrate Knowledge Graphs with LLMs (LLMs+KGs) generally outperform traditional methods, demonstrating that the extensive internal knowledge and advanced reasoning capabilities of LLMs are superior to specialized architectures. (2) LLMs+KGs methods consistently yield better performance than pure LLM strategies. This confirms that standalone LLMs are susceptible to hallucinations and lack strict alignment with structured facts, whereas KGs provide a reliable source of precise knowledge to ground the reasoning process. (3) COSMOS outperforms existing LLMs+KGs frameworks by explicitly optimizing for a globally interconnected subgraph rather than relying on disjoint relation paths. This superiority stems from our decoupled strategy: combining seed-guided local expansion with MaxST-based global aggregation ensures the retrieved evidence is both highly relevant and structurally coherent for complex multi-hop reasoning. (4) M^{3}GQA features multi-entity queries with long reasoning chains that pose significant challenges to existing methods. Notably, the granular retrieval performance evaluation (Table 9) demonstrates that COSMOS achieves higher evidence recall and answer accuracy than all baselines while using only half the retrieval budget of G-Retriever, validating the structural efficiency of our connectivity-optimized subgraph design.

5.3 Ablation Study

Impact of Different Components. To investigate the contribution of each component in COS-

Ablations	WebQSP		CWQ	
	Hit	F1	Hit	F1
w/o SGE	75.2 (-16.4)	67.6 (-10.6)	58.7 (-7.0)	46.4 (-9.2)
w/o TACA	90.1 (-1.5)	76.9 (-1.3)	63.4 (-2.3)	52.9 (-2.7)
w/o SACT	81.4 (-10.2)	66.2 (-12.0)	58.6 (-7.1)	49.1 (-6.5)
ours	91.6	78.2	65.7	55.6

Table 3: Impact of Different Components.

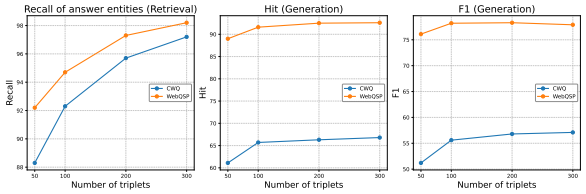

Figure 2: Impact of Retrieval Budget.

Backbones	WebQSP		CWQ	
	Hit	F1	Hit	F1
GPT-4o-mini	**91.6**	78.2	65.7	55.6
GPT-4o	**91.3**	**79.7**	**69.4**	**60.2**

Table 4: Impact of LLM Backbones.

MOS, we conduct an ablation study with three variants: (1) w/o Seed-Guided Greedy Expansion (SGE), (2) w/o Topology-Aware Component Aggregation (TACA), and (3) w/o Structure-Aware Contrastive Tuning (SACT).

The results summarized in Table 3, demonstrate that all three modules are essential for optimal performance, as removing any of them leads to a performance decrease. (1) Removing the SACT module leads to a sharp performance drop, as the dense retriever fails to bridge the structural gap between queries and the structured triplet format of KGs. Without this tuning, the model struggles to prioritize topologically relevant facts over those that are merely semantically similar, which is critical for identifying correct reasoning paths. (2) The absence of SGE causes a particularly substantial decrease in retrieval quality. SGE serves as the foundation of our framework by mining high-value local evidence centered around source entities. Its removal forces the model to attempt global connectivity without first securing a highly relevant set of local clusters, leading to a significant loss in information density. (3) Disabling TACA results in fragmented evidence subgraphs. While the local relevance of individual clusters is maintained, the lack of global connectivity prevents the LLM from traversing a complete reasoning chain. This confirms that resolving the connectivity constraint is necessary for the model to generate globally coherent and faithful reasoning results.

Impact of Retrieval Budget. Figure 2 examines the effect of the triplet budget $K \in \{50, 100, 200, 300\}$, reporting both (i) retrieval quality measured by the recall of answer entities and (ii) generation performance. As K increases from 50 to 100, we observe a sharp improvement in retrieval recall accompanied by a substantial gain in generation quality, indicating that a small budget constrains Seed-Guided Greedy Expansion to a narrow neighborhood and often fails to collect sufficient multi-hop evidence. Beyond $K = 100$, the gains gradually plateau, and at $K = 300$ the generation F1 on WebQSP slightly drops, suggesting information saturation: additional triplets increasingly introduce irrelevant or redundant facts that can distract the LLM during reasoning, a behavior also noted in prior retrieval-augmented frameworks. Considering both performance and API cost, we adopt $K = 100$ as a cost-effective operating point that strikes a favorable balance between evidence sufficiency and reasoning precision.

Impact of LLM Backbones. To evaluate the sensitivity of COSMOS to the underlying model, we compare its performance using GPT-4o-mini versus the more advanced GPT-4o. Our results, as demonstrated in Table 4, show a consistent performance gain when transitioning to the stronger backbone, indicating that the framework's effectiveness is further enhanced by superior base reasoning and synthesis capabilities. This improvement underscores that while COSMOS provides high-quality factual grounding to mitigate the lack of knowledge and hallucinations, the inherent capacity of the LLM to process and integrate complex evidence remains a critical determinant of final performance.

Transfer Tuning. To assess the generalization of our structure-aware contrastive tuning, we conduct transfer tuning experiments on WebQSP and CWQ, varying the source of fine-tuning data for the embedding model. For each target dataset, we compare three regimes: Target-only, where the embedding model is fine-tuned only on the target training set; Cross-dataset, where it is fine-tuned using the other dataset's training set; and Combined, where we fine-tune on the union of both training sets. Table 5 shows two consistent trends across both

Data Source	WebQSP		CWQ	
	Hit	F1	Hit	F1
Target	91.6	78.2	65.7	55.6
Cross	89.7	75.0	61.9	52.4
Combined	92.6	78.5	66.1	56.3

Table 5: Results of transfer tuning.

benchmarks. First, Combined achieves the best overall performance, suggesting that training on a broader distribution of queries and structures helps the retriever learn more general retrieval patterns. Second, Cross leads to only a moderate degradation compared to Target-only, indicating that our contrastive objective captures transferable structure-aware retrieval patterns rather than merely fitting dataset-specific artifacts. These results highlight the practicality of our tuning approach, especially when labeled data in the target domain is limited.

6 Conclusion

In this paper, we present COSMOS, a rigorous framework that reformulates subgraph retrieval for KBQA as a connectivity-constrained submodular maximization problem. This formulation marks a paradigm shift from heuristic path-finding to the active synthesis of evidence subgraphs with provable approximation guarantees. By employing a decompose-and-conquer strategy, COSMOS effectively resolves the dichotomy between local semantic relevance and global structural connectivity. Extensive experiments on WebQSP, CWQ, and M^3GQA demonstrate that COSMOS achieves state-of-the-art performance.

Ethical Consideration

We strictly adhere to the ACL Ethics Policy. Our experiments rely exclusively on publicly available datasets and publicly available API-based models, ensuring transparency and reproducibility without involving private or sensitive user data. We do not anticipate any direct negative social impacts arising from this work.

Limitations

While COSMOS establishes a rigorous framework for connectivity-oriented retrieval, we acknowledge five limitations. First, our current utility instantiation relies on the classic Maximal Marginal Relevance (MMR) criterion. Although MMR effectively captures the property of diminishing returns, it represents a relatively simple form of submodularity; future work could explore deep submodular functions parameterized by neural networks to better capture complex semantic dependencies. Second, the effectiveness of our Structure-Aware Contrastive Tuning relies on the availability of question-answer pairs for alignment. While we demonstrate transferability, future research may investigate unsupervised or few-shot alignment strategies to further reduce the dependency on domain-specific supervision. Third, the underlying connectivity-constrained submodular maximization is inherently challenging; COSMOS therefore adopts a two-stage decompose-and-conquer solver with stage-wise approximation guarantees. Developing a single-stage algorithm with a unified end-to-end approximation ratio remains an interesting direction for future work. Fourth, our current evaluation relies exclusively on standard Hit and F1 metrics, which may not fully capture reasoning fidelity and robustness against benchmark contamination; debate-driven evaluation paradigms offer a promising alternative for more rigorous future assessment (Cao and Zhao, 2025). Fifth, our current evaluation is limited to general-domain English benchmarks. Future work could extend COSMOS to two important directions: (1) multilingual settings, evaluating its performance on compositional relation reasoning tasks across diverse languages (Zhao and Zhang, 2024); and (2) specialized reasoning tasks such as temporal reasoning, which requires explicit modeling of time-dependent graph structures (Xiong et al., 2024).

Acknowledgments

This work is supported in part by Ucap Cloud.

References

Jinheon Baek, Alham Fikri Aji, Jens Lehmann, and Sung Ju Hwang. 2023a. Direct fact retrieval from knowledge graphs without entity linking. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2023, Toronto, Canada, July 9-14, 2023*, pages 10038–10055.

Jinheon Baek, Alham Fikri Aji, and Amir Saffari. 2023b. Knowledge-augmented language model prompting for zero-shot knowledge graph question answering.

Kurt D. Bollacker, Colin Evans, Praveen K. Paritosh, Tim Sturge, and Jamie Taylor. 2008. Freebase: a

collaboratively created graph database for structuring human knowledge. In *Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2008, Vancouver, BC, Canada, June 10-12, 2008*, pages 1247–1250.

Antoine Bordes, Nicolas Usunier, Sumit Chopra, and Jason Weston. 2015. Large-scale simple question answering with memory networks.

Linbo Cao and Jinman Zhao. 2025. Pretraining on the test set is no longer all you need: A debate-driven approach to QA benchmarks. In *Second Conference on Language Modeling*.

Abhimanyu Das and David Kempe. 2011. Submodular meets spectral: Greedy algorithms for subset selection, sparse approximation and dictionary selection. In *Proceedings of the 28th International Conference on Machine Learning, ICML 2011, Bellevue, Washington, USA, June 28 - July 2, 2011*, pages 1057–1064.

Mohammad Dehghan, Mohammad Ali Alomrani, Sunyam Bagga, David Alfonso-Hermelo, Khalil Bibi, Abbas Ghaddar, Yingxue Zhang, Xiaoguang Li, Jianye Hao, Qun Liu, Jimmy Lin, Boxing Chen, Prasanna Parthasarathi, Mahdi Biparva, and Mehdi Rezagholizadeh. 2024. EWEK-QA : Enhanced web and efficient knowledge graph retrieval for citation-based question answering systems. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2024, Bangkok, Thailand, August 11-16, 2024*, pages 14169–14187.

Guanting Dong, Rumei Li, Sirui Wang, Yupeng Zhang, Yunsen Xian, and Weiran Xu. 2023. Bridging the kb-text gap: Leveraging structured knowledge-aware pre-training for KBQA. In *Proceedings of the 32nd ACM International Conference on Information and Knowledge Management, CIKM 2023, Birmingham, United Kingdom, October 21-25, 2023*, pages 3854–3859.

Wenqi Fan, Yujuan Ding, Liangbo Ning, Shijie Wang, Hengyun Li, Dawei Yin, Tat-Seng Chua, and Qing Li. 2024. A survey on RAG meeting llms: Towards retrieval-augmented large language models. In *Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD 2024, Barcelona, Spain, August 25-29, 2024*, pages 6491–6501.

Nicholas Frosst, Nicolas Papernot, and Geoffrey E. Hinton. 2019. Analyzing and improving representations with the soft nearest neighbor loss. In *Proceedings of the 36th International Conference on Machine Learning, ICML 2019, 9-15 June 2019, Long Beach, California, USA*, pages 2012–2020.

Tiezheng Guo, Qingwen Yang, Chen Wang, Yanyi Liu, Pan Li, Jiawei Tang, Dapeng Li, and Yingyou Wen. 2024. Knowledgenavigator: Leveraging large language models for enhanced reasoning over knowledge graph.

Gaole He, Yunshi Lan, Jing Jiang, Wayne Xin Zhao, and Ji-Rong Wen. 2021. Improving multi-hop knowledge base question answering by learning intermediate supervision signals. In *WSDM '21, The Fourteenth ACM International Conference on Web Search and Data Mining, Virtual Event, Israel, March 8-12, 2021*, pages 553–561.

Xiaoxin He, Yijun Tian, Yifei Sun, Nitesh V. Chawla, Thomas Laurent, Yann LeCun, Xavier Bresson, and Bryan Hooi. 2024. G-retriever: Retrieval-augmented generation for textual graph understanding and question answering.

Boran Jiang, Yuqi Wang, Yi Luo, Dawei He, Peng Cheng, and Liangcai Gao. 2024. Reasoning on efficient knowledge paths: Knowledge graph guides large language model for domain question answering. In *IEEE International Conference on Knowledge Graph, ICKG 2023, Shanghai, China, December 1-2, 2023*, pages 142–149.

Jinhao Jiang, Kun Zhou, Zican Dong, Keming Ye, Xin Zhao, and Ji-Rong Wen. 2023a. Structgpt: A general framework for large language model to reason over structured data. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing, EMNLP 2023, Singapore, December 6-10, 2023*, pages 9237–9251.

Jinhao Jiang, Kun Zhou, Xin Zhao, Yang Song, Chen Zhu, Hengshu Zhu, and Ji-Rong Wen. 2025a. Kg-agent: An efficient autonomous agent framework for complex reasoning over knowledge graph. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pages 9505–9523.

Jinhao Jiang, Kun Zhou, Xin Zhao, and Ji-Rong Wen. 2023b. Unikgqa: Unified retrieval and reasoning for solving multi-hop question answering over knowledge graph. In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*.

Xinke Jiang, Ruizhe Zhang, Yongxin Xu, Rihong Qiu, Yue Fang, Zhiyuan Wang, Jinyi Tang, Hongxin Ding, Xu Chu, Junfeng Zhao, and Yasha Wang. 2025b. Hykge: A hypothesis knowledge graph enhanced RAG framework for accurate and reliable medical llms responses. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pages 11836–11856.

Bowen Jin, Chulin Xie, Jiawei Zhang, Kashob Kumar Roy, Yu Zhang, Zheng Li, Ruirui Li, Xianfeng Tang, Suhang Wang, Yu Meng, and Jiawei Han. 2024. Graph chain-of-thought: Augmenting large language models by reasoning on graphs. In *Findings of the Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and virtual meeting, August 11-16, 2024*, pages 163–184.

Jiho Kim, Yeonsu Kwon, Yohan Jo, and Edward Choi. 2023. KG-GPT: A general framework for reasoning on knowledge graphs using large language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023, Singapore, December 6-10, 2023*, pages 9410–9421.

Thomas N. Kipf and Max Welling. 2017. Semi-supervised classification with graph convolutional networks. In *5th International Conference on Learning Representations, ICLR 2017, Toulon, France, April 24-26, 2017, Conference Track Proceedings*.

Yunshi Lan, Gaole He, Jinhao Jiang, Jing Jiang, Wayne Xin Zhao, and Ji-Rong Wen. 2021. A survey on complex knowledge base question answering: Methods, challenges and solutions. In *Proceedings of the Thirtieth International Joint Conference on Artificial Intelligence, IJCAI 2021, Virtual Event / Montreal, Canada, 19-27 August 2021*, pages 4483–4491.

Yunshi Lan, Gaole He, Jinhao Jiang, Jing Jiang, Wayne Xin Zhao, and Ji-Rong Wen. 2023. Complex knowledge base question answering: A survey. *IEEE Trans. Knowl. Data Eng.*, 35(11):11196–11215.

Mufei Li, Siqi Miao, and Pan Li. 2025. Simple is effective: The roles of graphs and large language models in knowledge-graph-based retrieval-augmented generation. In *The Thirteenth International Conference on Learning Representations, ICLR 2025, Singapore, April 24-28, 2025*.

Zehan Li, Xin Zhang, Yanzhao Zhang, Dingkun Long, Pengjun Xie, and Meishan Zhang. 2023. Towards general text embeddings with multi-stage contrastive learning.

Guangyi Liu, Yongqi Zhang, Yong Li, and Quanming Yao. 2024. Explore then determine: A gnn-llm synergy framework for reasoning over knowledge graph.

Dan Luo, Jiawei Sheng, Hongbo Xu, Lihong Wang, and Bin Wang. 2023. Improving complex knowledge base question answering with relation-aware subgraph retrieval and reasoning network. In *International Joint Conference on Neural Networks, IJCNN 2023, Gold Coast, Australia, June 18-23, 2023*, pages 1–8.

Haoran Luo, Haihong E, Zichen Tang, Shiyao Peng, Yikai Guo, Wentai Zhang, Chenghao Ma, Guanting Dong, Meina Song, Wei Lin, Yifan Zhu, and Anh Tuan Luu. 2024a. Chatkbqa: A generate-then-retrieve framework for knowledge base question answering with fine-tuned large language models. In *Findings of the Association for Computational Linguistics, ACL 2024, Bangkok, Thailand and virtual meeting, August 11-16, 2024*, pages 2039–2056.

Linhao Luo, Yuan-Fang Li, Gholamreza Haffari, and Shirui Pan. 2024b. Reasoning on graphs: Faithful and interpretable large language model reasoning. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*.

Costas Mavromatis and George Karypis. 2025. GNN-RAG: graph neural retrieval for efficient large language model reasoning on knowledge graphs. In *Findings of the Association for Computational Linguistics, ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pages 16682–16699.

Alexander Miller, Adam Fisch, Jesse Dodge, Amir-Hossein Karimi, Antoine Bordes, and Jason Weston. 2016. Key-value memory networks for directly reading documents. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 1400–1409.

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

Barlas Oguz, Xilun Chen, Vladimir Karpukhin, Stan Peshterliev, Dmytro Okhonko, Michael Sejr Schlichtkrull, Sonal Gupta, Yashar Mehdad, and Scott Yih. 2022. Unik-qa: Unified representations of structured and unstructured knowledge for open-domain question answering. In *Findings of the Association for Computational Linguistics: NAACL 2022, Seattle, WA, United States, July 10-15, 2022*, pages 1535–1546.

Boci Peng, Yongchao Liu, Xiaohe Bo, Jiaxin Guo, Yun Zhu, Xuanbo Fan, Chuntao Hong, and Yan Zhang. 2025a. M^3gqa: A multi-entity multi-hop multi-setting graph question answering benchmark. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2025, Vienna, Austria, July 27 - August 1, 2025*, pages 30594–30620.

Boci Peng, Yongchao Liu, Xiaohe Bo, Sheng Tian, Baokun Wang, Chuntao Hong, and Yan Zhang. 2024. Subgraph retrieval enhanced by graph-text alignment for commonsense question answering. In *Machine Learning and Knowledge Discovery in Databases. Research Track - European Conference, ECML PKDD 2024, Vilnius, Lithuania, September 9-13, 2024, Proceedings, Part VI*, pages 39–56.

Boci Peng, Yun Zhu, Yongchao Liu, Xiaohe Bo, Haizhou Shi, Chuntao Hong, Yan Zhang, and Siliang Tang. 2025b. Graph retrieval-augmented generation: A survey. *ACM Trans. Inf. Syst.*, 44(2).

Stephen E. Robertson and Hugo Zaragoza. 2009. The probabilistic relevance framework: BM25 and beyond. *Found. Trends Inf. Retr.*, 3(4):333–389.

Apoorv Saxena, Adrian Kochsiek, and Rainer Gemulla. 2022. Sequence-to-sequence knowledge graph completion and question answering. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 2814–2828.

Apoorv Saxena, Aditay Tripathi, and Partha P. Talukdar. 2020. Improving multi-hop question answering

over knowledge graphs using knowledge base embeddings. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 4498–4507.

Xiangqing Shen, Fanfan Wang, and Rui Xia. 2025. Reason-align-respond: Aligning llm reasoning with knowledge graphs for kgqa.

Jiaxin Shi, Shulin Cao, Lei Hou, Juanzi Li, and Hanwang Zhang. 2021. Transfernet: An effective and transparent framework for multi-hop question answering over relation graph. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 4149–4158.

Haitian Sun, Tania Bedrax-Weiss, and William W. Cohen. 2019. Pullnet: Open domain question answering with iterative retrieval on knowledge bases and text. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 2380–2390.

Haitian Sun, Bhuwan Dhingra, Manzil Zaheer, Kathryn Mazaitis, Ruslan Salakhutdinov, and William W. Cohen. 2018. Open domain question answering using early fusion of knowledge bases and text. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 4231–4242.

Jiashuo Sun, Chengjin Xu, Lumingyuan Tang, Saizhuo Wang, Chen Lin, Yeyun Gong, Lionel M. Ni, Heung-Yeung Shum, and Jian Guo. 2024. Think-on-graph: Deep and responsible reasoning of large language model on knowledge graph. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*.

Alon Talmor and Jonathan Berant. 2018. The web as a knowledge-base for answering complex questions. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2018, New Orleans, Louisiana, USA, June 1-6, 2018, Volume 1 (Long Papers)*, pages 641–651.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. 2018. Graph attention networks.

Keheng Wang, Feiyu Duan, Sirui Wang, Peiguang Li, Yunsen Xian, Chuantao Yin, Wenge Rong, and Zhang Xiong. 2023a. Knowledge-driven cot: Exploring faithful reasoning in llms for knowledge-intensive question answering.

Wenhui Wang, Hangbo Bao, Shaohan Huang, Li Dong, and Furu Wei. 2021. Minilmv2: Multi-head self-attention relation distillation for compressing pretrained transformers. In *Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021, Online Event, August 1-6, 2021*, pages 2140–2151.

Xintao Wang, Qianwen Yang, Yongting Qiu, Jiaqing Liang, Qianyu He, Zhouhong Gu, Yanghua Xiao, and Wei Wang. 2023b. Knowledgpt: Enhancing large language models with retrieval and storage access on knowledge bases.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Yike Wu, Nan Hu, Sheng Bi, Guilin Qi, Jie Ren, Anhuan Xie, and Wei Song. 2023. Retrieve-rewrite-answer: A kg-to-text enhanced llms framework for knowledge graph question answering.

Siheng Xiong, Ali Payani, Ramana Kompella, and Faramarz Fekri. 2024. Large language models can learn temporal reasoning. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10452–10470.

Siheng Xiong, Ali Payani, Yuan Yang, and Faramarz Fekri. 2025. Deliberate reasoning in language models as structure-aware planning with an accurate world model. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 31900–31931.

Mufan Xu, Kehai Chen, Xuefeng Bai, Muyun Yang, Tiejun Zhao, and Min Zhang. 2025. Llm-based discriminative reasoning for knowledge graph question answering.

Rui Yang, Haoran Liu, Edison Marrese-Taylor, Qingcheng Zeng, Yuhe Ke, Wanxin Li, Lechao Cheng, Qingyu Chen, James Caverlee, Yutaka Matsuo, and Irene Li. 2024. Kg-rank: Enhancing large language models for medical QA with knowledge graphs and ranking techniques. In *Proceedings of the 23rd Workshop on Biomedical Natural Language Processing, BioNLP@ACL 2024, Bangkok, Thailand, August 16, 2024*, pages 155–166.

Wen-tau Yih, Matthew Richardson, Christopher Meek, Ming-Wei Chang, and Jina Suh. 2016. The value of semantic parse labeling for knowledge base question answering. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 2: Short Papers*.

Donghan Yu, Sheng Zhang, Patrick Ng, Henghui Zhu, Alexander Hanbo Li, Jun Wang, Yiqun Hu, William Yang Wang, Zhiguo Wang, and Bing Xiang. 2023. Decaf: Joint decoding of answers and logical forms for question answering over knowledge bases.

In *The Eleventh International Conference on Learning Representations, ICLR 2023, Kigali, Rwanda, May 1-5, 2023*.

Jing Zhang, Xiaokang Zhang, Jifan Yu, Jian Tang, Jie Tang, Cuiping Li, and Hong Chen. 2022. Subgraph retrieval enhanced model for multi-hop knowledge base question answering. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 5773–5784.

Jinman Zhao and Xueyan Zhang. 2024. Large language model is not a (multilingual) compositional relation reasoner. In *First Conference on Language Modeling*.

Kaiyan Zhao, Qiyu Wu, Xin-Qiang Cai, and Yoshimasa Tsuruoka. 2024a. Leveraging multi-lingual positive instances in contrastive learning to improve sentence embedding. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics, EACL 2024 - Volume 1: Long Papers, St. Julian's, Malta, March 17-22, 2024*, pages 976–991.

Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, Yifan Du, Chen Yang, Yushuo Chen, Zhipeng Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu, Peiyu Liu, Jian-Yun Nie, and Ji-Rong Wen. 2024b. A survey of large language models.

A Theorem Proof

A.1 Proof of Theorem 1

In this section, we provide the formal definition, intuitive interpretation, and detailed proof for Theorem 1.

Formal Definition and Interpretation of γ. Strictly speaking, the connectivity constraint in KGs prevents standard greedy algorithms from accessing the entire ground set at each step. We treat the Seed-Guided Greedy Expansion as an Approximate Greedy Algorithm, reference to (Das and Kempe, 2011).

Definition 1 (Topology-Semantic Alignment Factor γ)**.** *Let Ω be the global set of candidate edges. At step i, let $e^*_{global} \in \Omega$ be the edge that provides the maximum marginal gain globally, and let e^*_{local} be the edge selected by SGE from the current topological neighbors $\mathcal{N}(S_{i-1})$. We define $\gamma \in (0, 1]$ such that for any step i:*

$$\Delta F(e^*_{local} \mid S_{i-1}) \geq \gamma \cdot \Delta F(e^*_{global} \mid S_{i-1}). \quad (10)$$

The factor γ intuitively represents the smoothness of the reasoning landscape. A high γ implies that high-value evidence is topologically accessible via high-utility neighbors (i.e., no hidden gems blocked by low-score bridges). This theoretical insight motivates our Structure-Aware Contrastive Tuning module, which aims to maximize γ by aligning the embedding space with the graph structure.

Proof of Theorem 1. Next, we will give the complete proof of Theorem 1.

Proof. Let $\delta_i = F(S^*_{exp}) - F(S_i)$ denote the residual gain required to reach the optimal value at step i. Based on the submodularity and monotonicity of F, a standard result implies that there exists at least one edge in the optimal set S^*_{exp} that contributes at least δ_i/K to the objective.

Constrained by the local neighborhood, and by Definition 1, the gain of our locally selected edge e_i satisfies:

$$\begin{aligned} F(S_{i+1}) - F(S_i) &\geq \gamma \cdot \max_{e \in \Omega} \\ &\quad (F(S_i \cup \{e\}) - F(S_i)) \\ &\geq \gamma \cdot \frac{F(S^*_{exp}) - F(S_i)}{K}. \end{aligned} \quad (11)$$

Substituting δ_i, we derive the recurrence:

$$\delta_i - \delta_{i+1} \geq \frac{\gamma}{K}\delta_i \implies \delta_{i+1} \leq \left(1 - \frac{\gamma}{K}\right)\delta_i. \tag{12}$$

Applying this iteratively for K steps (with $\delta_0 = F(S^*_{exp})$):

$$\delta_K \leq \left(1 - \frac{\gamma}{K}\right)^K \delta_0 \approx e^{-\gamma} F(S^*_{exp}). \tag{13}$$

Substituting back $F(S_{SGE}) = F(S^*_{exp}) - \delta_K$, we obtain:

$$F(S_{SGE}) \geq (1 - e^{-\gamma}) F(S^*_{exp}). \tag{14}$$

□

A.2 Proof of Theorem 2

The TACA phase aims to connect disjoint components under a strict budget constraint. We analyze this process in two steps: (1) Structural Approximation, which maps the global search space to a tractably pruned Meta-Graph, and (2) Algorithmic Approximation, which applies greedy optimization on this Meta-Graph.

Definition 2 (Meta-Graph Fidelity Factor β)**.** *Let $OPT_{\mathcal{G}}$ be the utility of the optimal solution in the global Knowledge Graph, and $OPT_{\mathcal{M}}$ be the utility of the optimal solution identifiable within our constructed Meta-Graph $\mathcal{M}$ (constrained by hop limits and pruning). We define $\beta \in (0, 1]$ such that:*

$$F(OPT_{\mathcal{M}}) \geq \beta \cdot F(OPT_{\mathcal{G}}). \tag{15}$$

Proof of Theorem 2. Let S^*_{agg} denote the global optimal solution, so $F(S^*_{agg}) = F(OPT_{\mathcal{G}})$.

Step 1: Structural Bound. By Definition 2, the construction of the Meta-Graph (via strategies like Semantic PPR) retains a β-fraction of the optimal utility:

$$F(OPT_{\mathcal{M}}) \geq \beta \cdot F(S^*_{agg}). \tag{16}$$

Step 2: Algorithmic Bound. On the constructed Meta-Graph $\mathcal{M}$, COSMOS employs a greedy strategy (MaxST backbone followed by budgeted augmentation) to select the final edge set S_{TACA}. This is an instance of Monotone Submodular Maximization with Cardinality Constraints. According to the classical result by (Nemhauser et al., 1978), the greedy solution satisfies:

$$F(S_{TACA}) \geq \left(1 - \frac{1}{e}\right) \cdot F(OPT_{\mathcal{M}}). \tag{17}$$

Step 3: Synthesis. Combining the inequalities from Step 1 and Step 2:

$$\begin{aligned} F(S_{TACA}) &\geq \left(1 - \frac{1}{e}\right) \cdot F(OPT_{\mathcal{M}}) \\ &\geq \left(1 - \frac{1}{e}\right) \cdot \beta \cdot F(S^*_{agg}). \end{aligned} \tag{18}$$

Rearranging the terms yields the final bound:

$$F(S_{TACA}) \geq \beta \cdot \left(1 - \frac{1}{e}\right) \cdot F(S^*_{agg}). \tag{19}$$

□

B Computational Complexity and Efficiency Analysis

To rigorously evaluate the efficiency of COSMOS, we decouple the total inference latency T_{total} into two distinct components: (1) Structural Optimization Overhead (T_{struct}), which refers to the CPU-bound graph algorithms (SGE, TACA, PPR) executed locally; and (2) LLM Inference Overhead (T_{LLM}), which refers to the network-bound and computationally expensive calls to LLMs.

$$T_{total} = T_{struct} + N_{calls} \cdot T_{LLM\_unit}. \tag{20}$$

B.1 Structural Optimization Overhead (T_{struct})

This component represents the time consumed by the retrieval module. Let K be the total retrieval budget (e.g., $K = 100$), d_{avg} be the average node degree of the KG, and m be the number of identified source entities (which serves as the number of initial clusters).

Seed-Guided Greedy Expansion (SGE). In this phase, the algorithm iteratively selects the highest-scoring edges from the neighborhood frontier to expand the local subgraphs around the m source entities. At each iteration t (up to budget K_{SGE}), we evaluate the candidate edges in the current frontier. The size of the frontier is bounded by $\mathcal{O}(t \cdot d_{avg})$. Using a max-heap to manage candidate scores, the selection takes logarithmic time relative to the frontier size. The total complexity is $\mathcal{O}(K_{SGE} \cdot d_{avg} \cdot \log(K_{SGE} \cdot d_{avg}))$. Since K_{SGE} is a small constant constrained by the budget, this phase is highly efficient and independent of the global graph size $|\mathcal{V}|$.

Topology-Aware Component Aggregation (TACA). The goal is to bridge the m disjoint clusters. We compare our Semantic PPR approach against a naive enumeration baseline. Let $|V_{avg}|$ be the average number of nodes in each expanded cluster.

- **Naive Enumeration:** A brute-force strategy would attempt to find bridging paths between all pairs of nodes across all pairs of clusters. The complexity for pathfinding would be $\mathcal{O}(m^2 \cdot |V_{avg}|^2 \cdot d_{avg}^L)$. As $|V_{avg}|$ grows (e.g., dense local subgraphs), the quadratic term $|V_{avg}|^2$ makes this computationally prohibitive.

- **COSMOS (Semantic PPR):** By applying Semantic Personalized PageRank, we prune each of the m clusters to a small set of N representative nodes (e.g., $N = 3$) that are topologically central and semantically relevant. The pathfinding complexity reduces to $\mathcal{O}(m^2 \cdot N^2 \cdot d_{avg}^L + m \cdot T_{PPR})$.

Since N is a small constant ($N \ll |V_{avg}|$) and T_{PPR} is efficient on sparse graphs (linear to edges), COSMOS achieves a dramatic reduction in search space, ensuring that T_{struct} remains in the order of milliseconds.

B.2 LLM Inference Overhead (T_{LLM})

In modern KBQA systems, the bottleneck is predominantly the LLM inference latency (T_{LLM_unit}), which typically ranges from hundreds of milliseconds to seconds per call, orders of magnitude slower than CPU-based graph operations (T_{struct}). Therefore, minimizing the number of API calls (N_{calls}) is critical.

Our framework offloads the thinking process (navigation and pruning) to the efficient T_{struct} module. The LLM is invoked **only once** at the very end to generate the answer based on the retrieved subgraph. Conversely, approaches like ToG (Sun et al., 2024) require the LLM to evaluate neighbors at every hop, which results in an average of $N_{calls} \approx 10$ invocations in WebQSP and CWQ (Sun et al., 2024), and 30 calls in M^{3}GQA (Peng et al., 2025a), which leads to cumulative latency.

Since $T_{struct} \ll T_{LLM\_unit}$ (milliseconds vs. seconds), COSMOS achieves a speedup factor roughly proportional to the reduction in LLM calls, making it significantly more suitable for real-time applications than iterative search-based baselines.

C Algorithm Pseudocode

The retrieval and reasoning process of COSMOS is formally outlined in Algorithm 1.

Algorithm 1: COSMOS Framework

Require: Query Q, Knowledge Graph $\mathcal{G}$, Budget K, Encoder Φ.
Ensure: Generated Answer A.

Initialization: $V_Q \leftarrow \text{ExtractEntities}(Q)$, $q \leftarrow \Phi(Q)$.
// Seed-Guided Greedy Expansion (SGE)
$S_{local} \leftarrow \emptyset$, $\mathcal{C} \leftarrow \text{InitClusters}(V_Q)$.
while $|S_{local}| < K_{SGE}$ **do**
  $e^* \leftarrow \text{argmax}_{e \in \mathcal{N}(S_{local})} \Delta F(e \mid S_{local})$
  $S_{local} \leftarrow S_{local} \cup \{e^*\}$
end while
// Topology-Aware Component Aggregation (TACA)
Identify representatives R_i for each $C_i \in \mathcal{C}$ via Semantic PPR.
Construct Meta-Graph $\mathcal{M}$.
$\mathcal{T} \leftarrow \text{MaxST}(\mathcal{M})$
$S^* \leftarrow S_{local} \cup \text{Path}(\mathcal{T})$
// Budget-Aware Augmentation
while $|S^*| < K$ **do**
  $p^* \leftarrow \text{argmax}_{p \in \mathcal{P}_{cand}} \Delta F(p \mid S^*)$
  $S^* \leftarrow S^* \cup \{p^*\}$
end while
// LLM Reasoning
$A \leftarrow \text{LLM}(Q, \text{Linearize}(S^*))$
return A

D More Experimental Details

D.1 Data Statistics and Descriptions

All datasets used in our experiments are grounded in Freebase (Bollacker et al., 2008), which serves as the large-scale background knowledge graph containing approximately 88 million entities, 20,000 relations, and 126 million triplets. Table 6 and Table 7 summarize the data statistics.

WebQSP (Yih et al., 2016). This dataset primarily consists of questions requiring reasoning within 1 to 2 hops. It serves as a standard benchmark for evaluating the fundamental multi-hop reasoning capabilities of KBQA systems.

Dataset	#Train	#Validation	#Test
WebQSP	2,848	250	1,639
CWQ	27,639	3,519	3,531

Table 6: Data statistics of WebQSP and CWQ.

Setting	#Train	#Validation	#Test
Single-hop	925	154	463
Multi-hop	858	143	429
Set	802	133	400
Aggregation	681	114	341
Editing	555	93	278
Total	3,821	637	1,911

Table 7: Data statistics of M^3GQA.

CWQ (Talmor and Berant, 2018). Designed to be significantly more challenging than WebQSP, CWQ features intricate queries that require reasoning chains of up to 4 hops, rigorously testing the model's ability to navigate deep structural dependencies in the KG.

M^3GQA (Peng et al., 2025a). To further evaluate the robustness of COSMOS, we utilize M^3GQA, a highly challenging dataset designed to assess strong multi-hop reasoning and global planning capabilities. Queries in M^3GQA are characterized by high complexity, averaging over 3 entities per query, with reasoning paths generally exceeding 2 hops (excluding the single-hop setting). Unlike standard benchmarks, M^3GQA provides ground-truth retrieved subgraphs in addition to answers, enabling a direct assessment of retrieval fidelity. The dataset includes six distinct settings: single-hop, multi-hop, set, aggregation, editing, and answerability. In this work, we conduct retrieval and generation evaluations on the first five settings. We exclude the answerability setting, which focuses on detecting unanswerable queries, as it falls outside the scope of our current research on connectivity-oriented retrieval.

D.2 Details of Baselines

We compare our proposed COSMOS framework against a comprehensive set of baselines, ranging from traditional methods to recent LLM-integrated approaches. The details of these methods are as follows:

- **KV-Mem** (Miller et al., 2016): Utilize a Key-Value memory network to encode triplets, enabling multi-hop reasoning through iterative memory updates and read operations.
- **EmbedKGQA** (Saxena et al., 2020): Reformulate the multi-hop KBQA as a sequential link prediction problem, leveraging KG embeddings to model the interaction between queries and candidate answer entities.
- **NSM** (He et al., 2021): Employ a Neural State Machine that uses a sequential modeling approach to simulate the intermediate steps of multi-hop reasoning pathways over the knowledge graph.
- **TransferNet** (Shi et al., 2021): Operate on a GNN framework to infer answer entities by explicitly modeling the transfer of attention scores between the query and graph nodes.
- **KGT5** (Saxena et al., 2022): Adopt a generative paradigm by fine-tuning a T5 model directly on KG to produce answers given an input query.
- **GraftNet** (Sun et al., 2018): Combine entity linking with a heuristic subgraph retrieval mechanism to extract and reason over graph contexts that are relevant to the query.
- **PullNet** (Sun et al., 2019): Learn to construct query-specific subgraphs iteratively using an integrated retrieval module composed of an LSTM and a GNN.
- **SR+NSM** (Zhang et al., 2022): Enhance the reasoning process by first employing a relation-path retrieval module to filter relevant subgraphs before applying the Neural State Machine.
- **SR+NSM+E2E** (Zhang et al., 2022): An extension of SR+NSM that employs an end-to-end training strategy to jointly optimize the subgraph retrieval and downstream reasoning components.
- **KD-CoT** (Wang et al., 2023a): Leverage retrieved KG knowledge to guide LLMs in generating faithful CoT reasoning plans, improving interpretability.

- **UniKGQA** (Jiang et al., 2023b): Unify the traditionally separate stages of subgraph retrieval and reasoning into a single framework powered by LLMs.
- **Retrieve-Rewrite-Answer** (Wu et al., 2023): Decompose the retrieval process by first predicting the optimal reasoning depth and subsequently retrieving relation paths constrained by this predicted length.
- **G-Retriever** (He et al., 2024): Formulate subgraph retrieval as a Prize-Collecting Steiner Tree (PCST) optimization problem, aiming to identify a connected subgraph that maximizes semantic relevance while minimizing structural cost.
- **RoG** (Luo et al., 2024b): Generate faithful reasoning plans by training an LLM to produce logical relation chains, which are then grounded in the knowledge graph to extract valid reasoning paths that match the generated schema.
- **EtD** (Liu et al., 2024): Establish a synergistic framework between LLMs and GNNs (Explore then Determine), where the LLM predicts high-probability edges to guide the search, while the GNN handles the structural expansion of neighbor nodes.
- **SubgraphRAG** (Li et al., 2025): Enhance retrieval by incorporating a Directional Distance Encoding strategy to capture graph topology, utilizing these structural features to identify and select the most topologically significant edges for the evidence subgraph.

For M^{3}GQA, we add several baselines according to (Peng et al., 2025a), including:

- **Sparse**: Adopt the classic BM25 (Robertson and Zaragoza, 2009) algorithm as a sparse retriever to perform lexical matching, independently scoring and selecting the top-k triplets that share the highest keyword overlap with the input query.
- **Dense**: Utilize a dense retrieval paradigm where both the query and knowledge triplets are encoded into a shared vector space using a pre-trained embedding model, retrieving the top-k facts based on cosine similarity.
- **ToG** (Sun et al., 2024): Propose a Think-on-Graph strategy where an LLM acts as an autonomous agent to iteratively navigate the KG. Starting from the query entities, it dynamically evaluates and selects relevant edges and neighboring nodes hop-by-hop until sufficient information is gathered to answer the query.
- **RoG-SFT**: A variant of the RoG framework that undergoes additional Supervised Fine-Tuning (SFT) specifically on the target dataset.

D.3 Details of Implementations

Data Preprocessing. For the WebQSP and CWQ datasets, we strictly follow the standard protocol established by NSM (He et al., 2021) for entity linking and knowledge graph preprocessing. For the M^{3}GQA dataset, we utilize the pre-processed data provided by the original authors to ensure consistency with the benchmark settings.

Model Configuration. To facilitate a rigorous comparison, our model configurations are aligned with the baseline environments specific to each benchmark. For WebQSP and CWQ, we employ GPT-4o-mini as the backbone LLM and gte-large-en-v1.5 (Li et al., 2023) as the embedding model. Baseline results are taken from SubgraphRAG (Li et al., 2025) and RoG (Luo et al., 2024b) under their unified setting (matched LLM backbone, retrieval budget, and dataset pre-processing method), which we use unchanged for COSMOS. While for M^{3}GQA, we utilize GPT-4-Turbo as the reasoner and paraphrase-multilingual-MiniLM-L12-v2 (Wang et al., 2021) for encoding. For this benchmark, we report the reproduction results provided in the original paper (Peng et al., 2025a).

Hyperparameter Settings. For the retrieval strategy, the global retrieval budget is fixed at $K = 100$ triplets, partitioned into 90 for the Seed-Guided Greedy Expansion and 10 for Topology-Aware Component Aggregation. The parameter λ, controlling the trade-off between relevance and diversity in the utility function, is tuned from 0.4 to 0.9. For SGE, the layer-wise budget B_{layer} is selected from $\{15, 20, 30\}$. For Semantic PPR in TACA, the number of candidate representatives per component N is fixed to 3, and the temperature factor τ is set to 0.15. For training and inference, to ensure reproducibility, the temperature for all LLM

Baselines	Single-hop		Multi-hop		Set		Aggregation		Editing	
	Hit	F1	Hit	F1	Hit	F1	Hit	F1	Hit	F1
Traditional Methods										
Sparse	69.33	65.06	16.78	14.82	54.50	34.57	40.47	36.68	10.79	8.92
Dense	74.30	69.04	35.20	25.20	70.00	43.30	55.13	49.01	24.46	17.95
LLMs Methods										
GPT-4	35.85	33.41	12.82	11.87	44.75	21.92	26.03	25.65	-	-
ChatGPT	32.18	29.64	8.39	7.73	34.75	16.72	23.46	21.01	-	-
LLMs+KGs Methods										
ToG	74.00	71.33	20.00	19.00	44.00	23.13	30.00	28.67	20.00	20.00
RoG	61.12	57.05	19.58	16.78	50.75	27.94	39.59	35.72	11.51	8.19
RoG-SFT	73.00	68.54	20.75	17.10	58.00	31.14	36.36	34.01	12.59	8.82
G-Retriever	59.40	55.97	25.87	21.74	69.75	36.97	41.64	38.78	15.47	11.82
COSMOS w/o SACT	82.51	78.12	37.30	28.47	74.00	47.49	58.65	52.00	26.98	21.82
COSMOS (ours)	**87.90**	**83.01**	**41.49**	**31.06**	**79.75**	**51.90**	**59.82**	**53.54**	**33.81**	**24.83**

Table 8: Generation performance on M^3GQA.

Baselines	Single-hop			Multi-hop			Set			Aggregation			Editing		
	Recall	Acc	Num	Recall	Acc	Num	Recall	Acc	Num	Recall	Acc	Num	Recall	Acc	Num
Traditional Methods															
Sparse	41.60	79.27	100.00	32.68	24.01	100.00	26.90	39.00	100.00	23.92	59.24	100.00	31.84	19.42	100.00
Dense	67.73	88.55	100.00	51.34	66.43	100.00	46.50	53.75	100.00	45.53	82.99	100.00	51.35	59.71	100.00
LLMs+KGs Methods															
ToG	41.87	74.00	4.89	32.82	16.00	5.64	12.60	4.00	5.98	8.81	38.00	5.76	34.92	24.00	5.60
RoG	38.17	62.42	58.06	13.36	24.01	41.33	12.31	30.75	77.47	12.07	56.89	76.50	14.56	23.38	41.71
RoG-SFT	43.83	70.41	19.26	12.11	25.87	48.10	13.10	34.50	124.64	9.56	48.68	58.32	11.07	31.65	45.66
G-Retriever	50.15	82.94	208.53	41.72	**75.76**	236.10	29.44	60.50	207.25	31.88	79.47	199.60	40.43	67.99	226.91
COSMOS w/o SACT	78.96	**97.62**	100.00	57.52	67.83	100.00	49.44	84.53	100.00	47.08	91.79	100.00	56.02	61.87	100.00
COSMOS (ours)	**88.50**	96.98	100.00	**59.40**	71.10	100.00	**51.92**	**90.49**	100.00	**50.34**	**95.01**	100.00	**57.52**	**70.86**	100.00

Table 9: Retrieval performance on M^3GQA.

inference calls is set to 0. For the Structure-Aware Contrastive Tuning, we set the learning rate to 1e-5 and the batch size to 32. The model is trained for up to 5 epochs with an early stopping strategy based on validation set performance.

D.4 Generation and Retrieval Performance on M^3GQA

In this section, we present a comprehensive evaluation of COSMOS on the challenging M^3GQA benchmark, explicitly assessing both generation quality and retrieval fidelity. For generation, we retain the standard Hit and F1 Score as our primary performance indicators. For the retrieval evaluation, we adopt a more granular set of metrics to rigorously measure the quality of the evidence subgraphs: (1) Recall, which calculates the proportion of ground-truth triplets from the reasoning paths that are successfully retrieved; (2) Accuracy (Acc), which measures the percentage of questions for which all ground-truth answer nodes are present in the retrieved subgraph; and (3) Num, which reports the average number of triplets contained in the retrieved subgraphs to contextualize the information density relative to performance.

The comparative results for generation and retrieval are presented in Table 8 and Table 9, respectively. Analyzing the generation performance, baselines that demonstrated superior performance on WebQSP and CWQ, such as RoG and G-Retriever, exhibit a marked degradation on M^3GQA, surprisingly falling behind the simpler Dense Retrieval baseline in several metrics. This degradation highlights that heuristic or path-based methods struggle to satisfy multi-entity constraints simultaneously. In sharp contrast, COSMOS consistently achieves state-of-the-art performance across all five diverse settings. This substantial lead demonstrates the robustness of our framework and its superior capability in resolving complex, multi-constraint queries where traditional methods fail. Regarding retrieval performance, COSMOS maintains its dominance, outperforming baselines in the vast majority of scenarios. It is particularly noteworthy that COSMOS

surpasses G-Retriever even though the latter operates with a significantly larger retrieval budget (often double the size). This result highlights the structural efficiency of our approach: by optimizing for connectivity and global utility, COSMOS retrieves the necessary reasoning backbone more precisely without relying on excessive information retrieval.

D.5 Hyper-parameter Study

We conduct a hyperparameter sensitivity analysis to examine the robustness of our framework with respect to two key parameters: the trade-off coefficient λ in the MMR utility function, and the layer-wise budget B_{layer} used in greedy expansion. The experimental results are summarized in Figure 3.

Effect of λ in MMR. The parameter λ controls the trade-off between query relevance and diversity in the marginal utility function, balancing the similarity gain to the query against the redundancy penalty among selected edges. As shown in Figure 3, excessively small values of λ overemphasize diversity, which may suppress semantically consistent edges belonging to the same relation. In such cases, only a subset of edges associated with an important relation can be retrieved, leading to incomplete evidence coverage and degraded performance. In contrast, setting λ to relatively larger values (e.g., 0.8 or 0.9) yields the best results across datasets. This observation suggests that while query relevance should remain the dominant signal, explicitly incorporating a moderate diversity term in the utility function is beneficial for mitigating redundancy and improving the overall quality of the retrieved subgraph.

Effect of B_{layer} in greedy search. The parameter B_{layer} specifies the maximum number of edges that can be selected at each hop during the layer-wise greedy expansion. Figure 3 shows that model performance is relatively insensitive to variations of B_{layer} within a reasonable range. This indicates that the proposed greedy expansion strategy is robust to the precise per-layer budget allocation, as long as sufficient capacity is provided to capture salient local evidence. Such robustness is desirable in practice, as it reduces the need for careful tuning of this parameter across different datasets or query distributions.

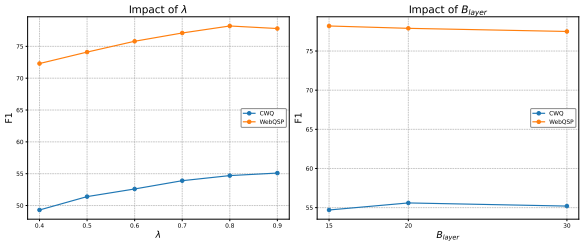

Figure 3: Hyper-parameter study.

D.6 Error Analysis

In this section, we randomly sample 50 incorrect predictions and analyze their error causes:

Missing numerical values in KG (38%). Some questions require explicit numerical values or numerical comparisons, but Freebase frequently stores numbers through mediated structures or machine IDs (e.g., "m." nodes) instead of directly readable literals. Consequently, even if COSMOS retrieves the structurally correct triplets, the evidence may not expose the required numeric value in a form usable by the LLM. This issue is prominent in queries such as *"Which countries does Japan export that have a GDP deflator change of rate -0.61?"*, where the numeric condition cannot be reliably verified from the retrieved subgraph.

Retrieval misses (30%). When COSMOS fails to retrieve one or more key triplets (e.g., due to long-range dependencies beyond hop constraints, low embedding similarity for critical bridge edges, or pruning during candidate path construction), the resulting evidence subgraph lacks the necessary reasoning backbone. In these cases, the LLM is forced to infer with insufficient evidence, and the final error should primarily be attributed to limited retrieval coverage rather than reasoning quality.

Correct retrieval but erroneous reasoning (22%). Even with complete supporting evidence, the LLM may produce incorrect outputs due to limited multi-step reasoning, prompt sensitivity, or failures to follow the required answer format. A representative example is *"In which years have the baseball club that claimed victory in the 1988 World Series won the World Series?"*, where the gold answers are entity strings such as *"1959 World Series; 1963 World Series; 1965 World Series; 1981 World Series; 1988 World Series"*, but the model outputs only bare years (e.g., *"1959; 1963; 1965; 1981; 1988"*). Under strict entity-level matching, such

non-canonical realizations are judged incorrect despite being semantically close.

Incomplete gold answers (8%). We observe cases where the dataset annotations appear incomplete for questions that naturally admit multiple valid answers, leading to the penalization of answers that are plausible but not listed in the gold set. For instance, *"What language is spoken in the country that circulates a newspaper called Manager Daily?"* resolves to Thailand, for which multiple languages may satisfy the predicate "spoken in the country." However, the gold annotation provides only a single language (e.g., *Khmer language*), creating false negatives under strict evaluation.

Query interpretation and answer-type mismatch (2%). A subset of failures stems from the incorrect interpretation of the expected answer type. For example, the query *"when was father chris riley born"* is temporally scoped, yet the system returns *"Echuca"*, which corresponds to a birthplace rather than a birth date. Such errors indicate that the downstream reasoner may violate the implicit type constraint induced by question semantics (e.g., *when* $\rightarrow$ date), particularly when the retrieved context contains salient but type-incompatible attributes.

E Prompts

The prompt used in the LLM reasoning phase is listed below.

The prompt for LLM reasoning.

Given a question and a relevant knowledge graph, please use one or more entities to answer the question.

Please first think step by step, and put the thinking process into the special tokens <think> and </think>. Then, output all entities (as comprehensive as possible) that you think are the correct answers, separate different entities with semicolons, and put them into the special tokens <answer> and </answer>.

It is necessary to ensure that the output entities actually exist in the provided knowledge graph. Complete output entity is required; abbreviations cannot be used.

If you believe that the answer cannot be inferred from the knowledge graph, then choose entities from the knowledge graph that are most likely to be the answers, or based on your own knowledge. At least one entity must be output between <answer> and </answer>, and the empty result cannot be output.

Question: {question}

Knowledge Graph: {retrieved subgraph}

Answer: