Constrained Structural Alignment: Mitigating Semantic Aliasing in RAG

Anonymous ACL submission

Abstract

Dense Retrieval-Augmented Generation (RAG) often falters in complex multi-hop reasoning due to semantic aliasing, where retrieved contexts are topically similar but logically inconsistent with the query's relational intent. We identify this as a fundamental semantic-structural gap: while vector similarity is inherently symmetric, logical reasoning requires the preservation of asymmetric predicates. To resolve this, we present CoSA-RAG, an exploratory, high-precision graph verification framework that reformulates retrieval verification as a constrained structural alignment problem. Operating as a plug-and-play verification layer on top of dense retrievers, CoSA-RAG transforms raw linguistic dependencies into a directed logical backbone through a schema-constrained mapping to enforce strict directionality. By leveraging an Edge Compatibility Kernel, our method rigorously prunes relational noise that defies the query's underlying logic before context is fed to the generator. Extensive evaluations on HotpotQA, MusiqueQA, and 2WikiMultihop demonstrate that as a filtering layer, CoSA-RAG consistently refines retrieved contexts, outperforming iterative reasoning frameworks like IRCoT by up to 14.8 points EM while significantly reducing latency and token consumption. Our work provides a practical, high-precision alternative for mitigating structural hallucinations in robust RAG systems.

1 Introduction

Retrieval-Augmented Generation (RAG) effectively mitigates knowledge cutoffs and hallucinations in Large Language Models (LLMs) (Lewis et al., 2020; Gao et al., 2023). By grounding generation in external corpora, RAG combines reasoning with factual precision. While dense retrievers excel at factoid lookups, they falter on complex multi-hop reasoning requiring logical deduction (Trivedi et al., 2023; Mavi et al., 2024). The bottleneck lies not in the *quantity* of retrieved information, but in the *logical alignment* between query intent and evidence.

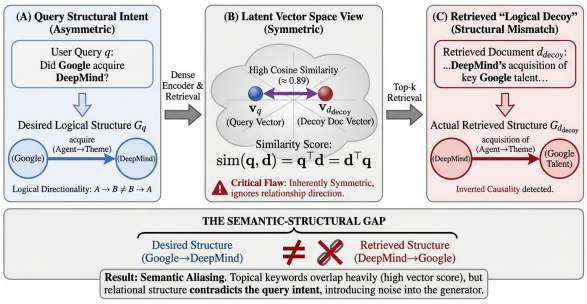

Figure 1: Illustration of the Semantic-Structural Gap causing Semantic Aliasing in Dense Retrieval RAG.

We formally define this limitation as the ***Semantic-Structural Gap***. Contemporary retrievers map queries and documents into a shared latent space, causing a geometric mismatch: vector similarity is inherently symmetric ($\mathbf{q} \cdot \mathbf{d} = \mathbf{d} \cdot \mathbf{q}$), whereas reasoning predicates are asymmetric and directional (e.g., $Subject \xrightarrow{Agent} Object \neq Object \xrightarrow{Agent} Subject$). Consequently, dense retrievers suffer from *semantic aliasing* (Karpukhin et al., 2020), retrieving structurally inconsistent but topically overlapping documents ("logical decoys") with high confidence. For example, given *"Did Google acquire DeepMind?"*, bi-encoders highly rank a decoy document stating *"DeepMind's acquisition of key Google talent..."*, ignoring the inverted causality. This relational noise degrades model faithfulness (Cuconasu et al., 2024).

Recent advancements address this via two streams. *Iterative reasoning* (e.g., IRCoT (Trivedi et al., 2023)) interleaves retrieval and generation but incurs prohibitive latency from multiple LLM calls. Conversely, *Graph-enhanced RAG* models corpora as entity-relation networks (Edge et al., 2024; Pan et al., 2024) but focuses on *context expansion* via neighborhood retrieval. By neglecting *verification*, these methods fail to validate if the retrieved topology matches the query's logical

structure, leading to a *granularity-noise dilemma* where rigid matching yields low recall, while unconstrained expansion overwhelms the generator.

To resolve this, we propose **CoSA-RAG** (**Co**nstrained **S**tructural **A**lignment **RAG**), a training-free, post-retrieval graph verification framework. Instead of expanding context, CoSA-RAG enforces structural constraints to prune logical decoys *before* generation. To ensure efficiency, we employ a synergistic offline-online pipeline: offline, we index the corpus into document graphs; online, we model the query as a *Relational Semantic Graph* (RSG) via a dual-granularity module that bridges entity anchors and abstract concepts. We then perform *Adaptive Isomorphic Pruning* on candidate contexts, replacing rigid matching with a soft *Maximum Common Subgraph* (MCS) metric grounded in edge compatibility kernels. To avoid heuristic thresholding, we ground pruning decisions in *Extreme Value Theory (EVT)*, dynamically scaling filter strictness based on query entropy.

Our contributions are summarized as follows:

- We formalize the *Semantic-Structural Gap*, attributing multi-hop retrieval failures to the symmetry of vector metrics applied to asymmetric logical relations.
- We introduce **CoSA-RAG**, a training-free, plug-and-play graph verification framework synergizing dual-granularity modeling with soft subgraph matching, supported by EVT-grounded theoretical analysis for adaptive noise filtering.
- Empirical evaluations on three complex multi-hop benchmarks demonstrate that CoSA-RAG yields consistent, substantial accuracy improvements—delivering up to **21.1 points EM gain** over contemporary graph-RAG baselines and outperforming iterative reasoning frameworks like IRCoT by up to **14.8 points EM**—while reducing generator token consumption by approximately 23% via effective structural pruning.

2 Preliminaries

We formalize Retrieval-Augmented Generation (RAG) as a *constrained structural alignment* task rather than a pure semantic search problem. Let $\mathcal{D} = \{d_1, \ldots, d_N\}$ denote the document corpus, and q be the user query.

2.1 Probabilistic RAG Framework

Standard RAG approximates the generation probability $P(y|q)$ by marginalizing over retrieved documents z from the entire corpus $\mathcal{D}$. However, performing structural verification over millions of documents is computationally intractable. To resolve this, we formulate RAG as a two-stage process.

Let $Z_q \subset \mathcal{D}$ (where $|Z_q| = K \ll N$) denote a candidate set of top-K documents retrieved by a dense retriever $P(z|q)$. We refine the generation probability by introducing a binary *structural consistency variable* $S \in \{0, 1\}$, conditioning generation on both topical relevance and structural validity over the candidate set:

$$P(y|q) \approx \sum_{z \in Z_q} P(y|z, q) \underbrace{P(S = 1|G_q, G_z)}_{\text{Structural Verification}} P(z|q), \tag{1}$$

where G_q and G_z represent the topological semantic structures of the query and the candidate document, respectively.

2.2 Graph-Theoretic Representation

To capture relational semantics, we project textual segments into a unified topological space $\mathcal{G}$.

Definition 1 (Relational Semantic Graph)**.** *A text segment $t \in \{q, z\}$ is represented as a directed, relation-labeled graph $G_t = (V_t, E_t)$ over a closed set of logical predicates $\mathcal{L}$. Here, V_t represents the set of explicit entities or abstract concepts extracted from t, and $E_t \subseteq V_t \times \mathcal{L} \times V_t$ denotes the set of directional relational triples (u, r, v) where $r \in \mathcal{L}$.*

To bridge symbolic structures and continuous semantics, we define node and relation embedding functions $\phi : V_t \to \mathbb{R}^h$ and $\psi : \mathcal{L} \to \mathbb{R}^h$ to map vertices and predicates into low-dimensional vector spaces, enabling soft topological alignment.

2.3 The Semantic-Structural Gap

Let $f_\theta(q, z)$ be a standard dense retrieval score. We define *Semantic Aliasing* as the scenario where a retrieved document $z \in Z_q$ satisfies the dense retrieval threshold δ but violates the structural consistency threshold τ:

$$f_\theta(q, z) \geq \delta \quad \wedge \quad \text{MCS}(G_q, G_z) < \tau(q), \tag{2}$$

where $\text{MCS}(\cdot, \cdot)$ quantifies the soft Maximum Common Subgraph similarity, and $\tau(q)$ is an adaptive threshold. Documents satisfying Eq. 2 constitute *relational noise*—topically congruent but logically incoherent decoys.

3 Methodology

Building upon the probabilistic formulation in Eq. 1, CoSA-RAG estimates the structural consistency probability $P(S = 1|G_q, G_z)$ through a three-phase pipeline: (1) Structural Intent Modeling, (2) Soft-Schema Graph Projection, and (3) Adaptive Isomorphic Pruning.

3.1 Phase I: Structural Intent Modeling

We model the query q as a Relational Semantic Graph $G_q = (V_q, E_q)$. Unlike standard keyword extraction which treats terms independently, our approach aims to reconstruct the logical backbone of the user's inquiry. To ensure robustness against terminological variations while preserving strict semantic directionality, we employ a **dual-granularity intent modeling** strategy implemented via a constrained semantic parsing pipeline.

Dual-Granularity Node Representation. Following the dual-level representation philosophy of LightRAG (Guo et al., 2024), we define the node set as $V_q = V_s \cup V_a$ to bridge the semantic gap between explicit entity anchors and abstract conceptual queries. Here, V_s and V_a represent distinct semantic granularities:

- V_s: **Specific entities** (e.g., "DeepMind", "AlphaGo"). These serve as rigid topological anchors, ensuring that the retrieved contexts remain strictly grounded in the specific physical entities requested by the query.
- V_a: **Abstract concepts** (e.g., "acquisition strategy", "market impact"). These nodes capture high-level relational intent beyond surface forms, enabling the alignment layer to match semantically equivalent but lexically divergent formulations.

Crucially, while LightRAG (Guo et al., 2024) treats these dual granularities primarily as expansion signals for broadening retrieval recall, we repurpose them as *verification anchors* to enforce structural consistency during alignment.

To resolve polysemy, we employ a *query-conditioned* embedding function $\phi : V_q \to \mathbb{R}^h$ using a RoBERTa-based encoder, contrasting with static Knowledge Graph Embeddings (KGE) that assign fixed vectors. For a node v spanning specific character indices in q:

$$\phi(v) = \text{Pool}\left(\text{Encoder}(q)_{\text{indices}(v)}\right). \quad (3)$$

Logical Predicate ($\mathcal{L}$)	Syntactic Source	Semantics Description
AGENT	`nsubj`, `agent`	The initiator of an action.
THEME	`dobj`, `nsubjpass`	The entity undergoing the action.
CONDITION	`advcl` (if, when)	Logical preconditions or temporal scope.
CAUSE	`advcl` (because), `obl`	Causal relationships between events.
ATTRIBUTE	`amod`, `appos`	Properties or definitions of an entity.
COMPARE	`conj` (vs, than)	Comparative or contrastive relations.

Table 1: The mapping schema $\mathcal{M}$ from linguistic dependencies to Logical Predicates $\mathcal{L}$. This taxonomy ensures that structural intent is preserved during graph construction.

This yields context-specific representations (e.g., distinguishing "Apple" as a company from the fruit) to prevent false-positive matching during pruning.

Logical Dependency Mapping. The edge set E_q constitutes the relational scaffolding that binds discrete entities into a coherent reasoning chain. A critical deficiency in standard dense retrieval is its inherent *symmetry blindness*, where symmetric vector metrics fail to distinguish between active and passive logical roles (e.g., confusing "*A acquires B*" with "*B acquires A*"). Although syntactic dependency parsing introduces directionality, it produces overly granular labels that dilute the semantic signal and complicate graph matching.

To mitigate this, we define a projection $\mathcal{M} : \mathcal{T}_{syn} \to \mathcal{T}_{log}$ that transforms raw linguistic dependencies into a concise inventory of high-level logical predicates $\mathcal{L}$ (detailed in Table 1). This projection converts the query into a directed logical backbone, enforcing strict asymmetric constraints.

In practice, to ensure robustness against ungrammatical inputs, we implement this mapping via a **schema-constrained instruction prompt** fed to a lightweight LLM (GPT-4o-mini) acting as a unified semantic parser. As shown in our prompt design (Appendix D), the model extracts relational triples $(u, r, v) \in E_q$ where $u, v \in V_q$ and $r \in \mathcal{L}$ in a single-shot execution, ensuring that the resulting node categories (V_s, V_a) and directed edge set E_q strictly conform to our formal graph ontology.

3.2 Phase II: Soft-Schema Graph Projection

Graph-based RAG often suffers from *schema mismatch*: queries are typically abstract and high-level, whereas documents contain granular, diverse linguistic details. This vocabulary gap renders rigid exact-matching ineffective, missing legitimate semantic connections due to lexical discrepancies.

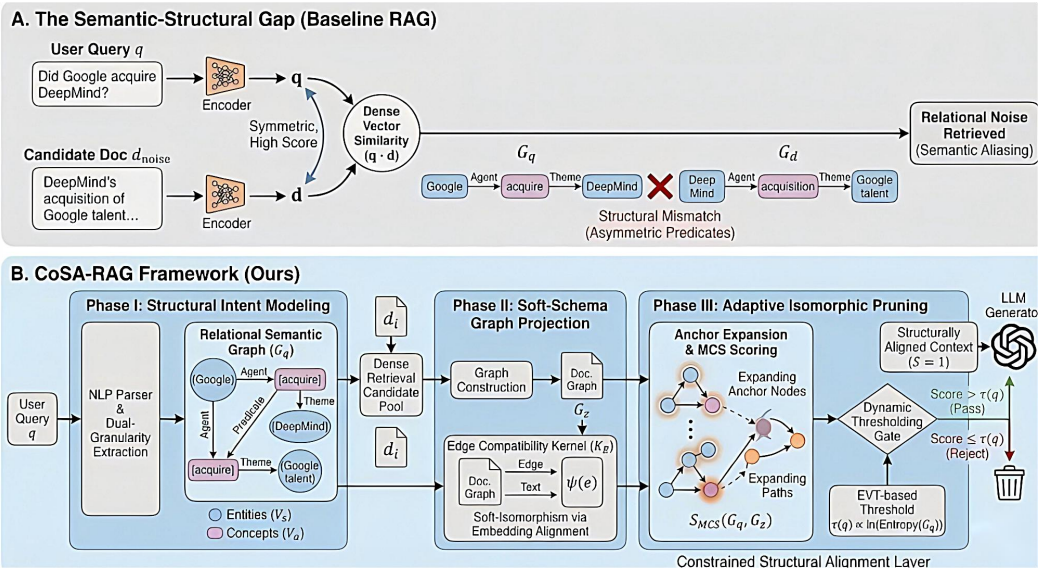

Figure 2: Illustration of the semantic-structural gap in conventional RAG and our CoSA-RAG framework. (A) Standard dense retrieval suffers from semantic aliasing: symmetric similarity metrics conflate topically related but logically inconsistent relations. (B) CoSA-RAG bridges this gap through a three-stage structural alignment pipeline: (i) Dual-granularity intent modeling; (ii) Soft-schema pattern projection; (iii) EVT-guided isomorphic pruning to dynamically threshold alignment scores.

To bridge this, we propose a **Soft-Isomorphism** mechanism grounded in Reproducing Kernel Hilbert Spaces (RKHS). To ensure computational viability during online inference, document graphs G_z (where $z \in Z_q$) are constructed and stored offline during the corpus indexing phase using the same schema-constrained pipeline as G_q. Under our triple-based representation, we formally define the alignment between a query relation $r_q \in \mathcal{L}$ and a candidate document relation $r_z \in \mathcal{L}$ via an **Edge Compatibility Kernel** $K_E : \mathcal{L} \times \mathcal{L} \to [0, 1]$:

$$K_E(r_q, r_z) = \underbrace{\langle \Psi(r_q), \Psi(r_z) \rangle_{\mathcal{H}}}_{\text{Semantic Term}} \cdot \underbrace{\mathbb{I}[\text{dir}(e_q) \cong \text{dir}(e_z)]}_{\text{Structural Term}} \tag{4}$$

Here, $\Psi(\cdot)$ is a feature map projecting symbolic relation labels into a semantic vector space $\mathcal{H}$, approximated by a Sentence-BERT encoder fine-tuned on NLI tasks. The indicator $\mathbb{I}[\cdot]$ enforces a strict *directionality constraint* over the mapped edges. Specifically, for a query edge $e_q = (u, r_q, u') \in E_q$ aligned to a document edge $e_z = (v, r_z, v') \in E_z$ under a node assignment matrix $\mathbf{P}$ (i.e., $\mathbf{P}_{uv}\mathbf{P}_{u'v'} = 1$), the indicator returns 1 if the agentive or causal flow is preserved (e.g., $u \xrightarrow{r_q} u'$ is structurally compatible with $v \xrightarrow{r_z} v'$), and 0 if the direction is inverted (e.g., mapping to $v' \xrightarrow{r_z} v$). This formulation relaxes the discrete graph isomorphism problem into a continuous optimization space, tolerating lexical variations while rigorously penalizing directional contradictions.

3.3 Phase III: Adaptive Isomorphic Pruning

To verify structural consistency, we seek to identify the Maximum Common Subgraph (MCS) between G_q and G_z. Mathematically, this corresponds to solving a **Quadratic Assignment Problem (QAP)** to find a node-to-node alignment matrix $\mathbf{P} \in \{0, 1\}^{|V_q| \times |V_z|}$ that maximizes the structural similarity objective:

$$J(\mathbf{P}) = \sum_{u \in V_q} \sum_{v \in V_z} \mathbf{P}_{uv} \cos\big(\phi(u), \phi(v)\big) + \lambda \sum_{(u, r_q, u') \in E_q} \sum_{(v, r_z, v') \in E_z} \mathbf{P}_{uv}\mathbf{P}_{u'v'} K_E\big(r_q, r_z\big), \tag{5}$$

subject to the *partial injectivity constraint*: $\sum_v \mathbf{P}_{uv} \leq 1, \forall u$ and $\sum_u \mathbf{P}_{uv} \leq 1, \forall v$.

Entity-Anchored Topological Propagation. Directly solving QAP is NP-hard due to the combinatorially chaotic matching space of polysemous semantic concepts. However, we exploit the unique topological asymmetry of Relational Semantic Graphs to design a polynomial-time propagation heuristic: **Entity-Anchored Topological Propagation** (Appendix. E).

The physical intuition relies on the semantic uniqueness of specific physical entities (V_s), which

are highly unambiguous in vector spaces and yield high-confidence, near-unique 1-to-1 matching alignments. Our algorithm first aligns these "Specific Entities" as *seeds/anchors* (Line 3). Once these robust anchor points are locked, we propagate the alignment constraints along the directed edges to their neighborhood sets (Line 9-15). This bottom-up propagation effectively collapses the matching search space and propagates semantic certainty, resolving the matching of abstract, highly polysemous concepts (V_a).

The final score is normalized by the query's structural volume ($|V_q| + \lambda|E_q|$), reflecting our *Recall-oriented* design: we measure how well the document *covers* the query's requisite logic, rather than penalizing extra context in the document.

Complexity Analysis. The computational bottleneck lies in sorting candidate node pairs, with a complexity of $O(|V_q||V_z| \log(|V_q||V_z|))$. The propagation step performs a localized search bounded by the maximum degree Δ. Since semantic graphs are sparse ($\Delta \ll |V|$), the overall complexity is approximately quadratic $O(|V_q||V_z|)$, which is computationally negligible compared to LLM inference latency, ensuring high-efficiency real-time retrieval.

3.4 Theoretical Analysis: Dynamic Thresholding with Extreme Value Theory

Using a static threshold for S_{MCS} is ineffective as it ignores the variance in query complexity. We instead ground our decision boundary in **Extreme Value Theory (EVT)**.

Handling Structural Dependencies. A technical challenge in applying EVT is the violation of the independent and identically distributed (i.i.d.) assumption, as subgraph matching scores exhibit spatial correlation due to shared nodes. However, we rely on Extreme Value Theory for weakly dependent sequences (Leadbetter, 1974), which establishes that the maximum of a stationary sequence still converges to a Gumbel distribution provided that the long-range dependence is weak (specifically, satisfying Leadbetter's $D(u_n)$ mixing condition). In our RAG scenario, semantic graphs are sparse, and structural noise typically manifests as local mismatches rather than global isomorphism. Consequently, the spatial correlation decays exponentially with graph distance, allowing the alignment scores to be modeled as *Block Maxima* of weakly dependent variables.

Noise Distribution & Threshold. We model the distribution of alignment scores S_{noise} under the Null Hypothesis H_0 of no structural correlation between the retrieved document and the query. As the alignment score represents a maximum over local subgraph matches, it converges to a Gumbel distribution:

$$P(S_{\text{noise}} \leq s) = \exp\left(-\exp\left(-\frac{s-\mu}{\beta}\right)\right). \quad (6)$$

To enforce a constant False Positive Rate (FPR) α, we define the decision boundary τ such that $P(S_{\text{noise}} > \tau) = \alpha$. Drawing on random graph theory (Bollobás, 2011), the expected size of a maximum common subgraph between a query graph G_q and an uncorrelated noise graph scales logarithmically with the query graph size. Thus, the location parameter μ is proportional to the combinatorial entropy of the query's topological size, yielding the dynamic threshold function:

$$\tau(q) = \gamma \cdot \ln(1 + |V_q| + |E_q|) + \delta. \quad (7)$$

This formulation dictates that queries with higher structural complexity require strictly higher evidence scores to reject the Null Hypothesis.

Information-Theoretic Interpretation. Eq. 7 provides a statistical justification for adaptive pruning based on combinatorial entropy. As the query graph grows in complexity, the combinatorial search space for partial subgraphs expands exponentially, increasing the probability of observing spurious matches within random noise. Therefore, maintaining a fixed significance level α necessitates a threshold that scales with the query's information content. This adaptive mechanism allows CoSA-RAG to distinguish genuine structural isomorphism from stochastic overlaps, effectively filtering out "logical decoys" that exhibit high keyword density but lack topological coherence.

4 Experiments

Our experimental evaluation is designed to answer three core research questions:

RQ1 (Effectiveness): Does CoSA-RAG effectively mitigate semantic aliasing and improve reasoning accuracy compared to single-turn baselines?

RQ2 (Theoretical Validity): Do the empirical distributions of structural noise and optimal thresholds align with our derivations based on Extreme Value Theory?

Model	HotpotQA				MuSiQue				2Wiki			
	R@2	R@5	EM	F1	R@2	R@5	EM	F1	R@2	R@5	EM	F1
Standard & Dense Retrieval Baselines												
BM25	55.4	72.5	–	–	32.3	41.2	–	–	51.8	61.9	–	–
Contriever	57.2	75.5	–	–	34.8	46.6	–	–	56.6	57.5	–	–
ColBERTv2	61.2	76.8	–	–	36.5	47.9	–	–	57.4	65.5	–	–
BGE-M3	64.7	79.3	–	–	37.9	49.2	–	–	59.2	68.2	–	–
Graph / LLM-Based Retrieval Baselines												
GPT-4o-mini	–	–	30.4	42.8	–	–	12.5	24.1	–	–	31.0	39.0
GraphRAG	–	–	35.2	46.8	–	–	14.8	25.4	–	–	38.5	44.2
LightRAG	–	–	36.8	48.3	–	–	18.1	27.5	–	–	45.1	49.5
IRCoT	58.6	73.8	43.1	55.8	34.9	46.0	20.5	28.8	63.9	73.5	46.8	52.5
HippoRAG	63.0	77.3	51.8	63.4	44.8	51.9	22.2	34.9	69.2	88.4	60.5	71.8
HippoRAG2	**88.7**	**94.3**	**63.1**	**75.5**	67.5	74.7	**37.2**	**48.6**	**81.9**	**90.4**	59.4	71.0
CoSA-RAG (Ours)	78.1	89.4	57.9	68.2	**68.7**	**76.0**	34.3	44.5	70.1	88.8	**62.1**	**73.3**

Table 2: Unified evaluation results on multi-hop QA benchmarks, presenting both retrieval quality (Recall@K under $K_{\text{init}} = 20$) and downstream generation accuracy (EM/F1) side-by-side. Category divider rows group methods by model paradigm. Underlined values indicate the best results within each category, while **bolded** values denote the absolute best results across all evaluated single-turn methods. Hyphens (–) denote concurrent baseline results or non-applicable generation scores.

RQ3 (Robustness & Efficiency): Is the framework robust to graph construction errors and computationally efficient for real-time deployment?

4.1 Experimental Setup

Datasets. We utilize three multi-hop reasoning benchmarks specifically selected for their high density of *semantic aliasing* (i.e., distractors that are topically relevant but logically inconsistent): **HotpotQA** (Yang et al., 2018), **MuSiQue** (Trivedi et al., 2022), and **2WikiMultihopQA** (Ho et al., 2020).

Baselines. We compare against three categories of methods: (1) **Standard Retrieval:** BM25, Contriever (Izacard et al., 2022), BGE-M3 (Chen et al., 2024), and ColBERTv2 (Santhanam et al., 2022). (2) **Graph/LLM-based RAG:** GraphRAG (Edge et al., 2024), HippoRAG (Jimenez Gutierrez et al., 2024), HippoRAG2 (Gutiérrez et al., 2025), and LightRAG (Guo et al., 2024). Note that LightRAG is our primary baseline as it shares the dual-keyword representation but lacks structural verification. (3) **Iterative Reasoning:** IRCoT (Trivedi et al., 2023). While IRCoT is an iterative method (unlike our one-shot approach), we include it to benchmark reasoning capability.

Implementation. To ensure a strictly fair and controlled evaluation, all baseline methods and CoSA-RAG are unified under the same model configurations: we retrieve an initial candidate set Z_q of size $K_{\text{init}} = 20$ using a unified dense retriever (Contriever (Izacard et al., 2022)). All downstream generators are unified using the `GPT-4o-mini` API with a temperature of 0.0. We use `roberta-large-mnli` (Liu et al., 2019) for the Soft-Schema Kernel (K_E). The hyperparameter λ in Eq. 5 is set to 0.5.

4.2 Main Results (RQ1)

Before analyzing downstream reasoning, we assess both retrieval and generation efficacy in Table 2.

Retrieval Performance Analysis. As shown in the retrieval columns of Table 2, CoSA-RAG consistently outperforms its primary baseline, LightRAG, across all benchmarks, particularly on 2WikiMultihopQA. Crucially, although CoSA-RAG acts as a post-retrieval filter over a fixed candidate pool ($K_{\text{init}} = 20$), it significantly improves the final Recall@5 (e.g., from 76.5% to 89.4% on HotpotQA). This recall improvement stems from our structural verification layer, which successfully prunes highly-ranked "logical decoys" from the dense retriever's output, allowing true relevant con-

texts originally ranked lower (e.g., between positions 6 and 20) to rise into the final top-5 slots. While HippoRAG slightly outperforms our method in 2WikiMultihopQA recall, its global Personalized PageRank and associative memory expansion are explicitly designed for broad topological *expansion*, maximizing recall at the expense of increased context size. In contrast, CoSA-RAG operates as a high-precision filter, prioritizing the rigorous removal of structural noise while maintaining highly competitive recall.

Comparison with Iterative Methods. Our one-shot approach consistently outperforms iterative frameworks like IRCoT across all datasets by up to 14.8 EM points (on HotpotQA) while significantly reducing latency (Sec. 4.4). This suggests that single-step verification offers a highly efficient alternative to costly iterative reasoning chains.

Response Quality. As shown in the generation columns of Table 2, CoSA-RAG establishes a strong performance baseline for single-turn graph-based methods, yielding a substantial average improvement of 18.1 points EM over LightRAG.

Analysis of the Retrieval-Response Gap. An interesting discrepancy arises on the MuSiQue dataset: while CoSA-RAG achieves the highest retrieval recall (R@2 of 68.7), its final response accuracy (EM of 34.3) is lower than HippoRAG2 (37.2). This phenomenon reveals a fundamental trade-off in single-document topological verification. In highly fragmented multi-hop datasets like MuSiQue, the reasoning path is distributed across disjoint documents. Because CoSA-RAG evaluates structural isomorphism on individual candidate documents, it may aggressively prune partially matched contexts that are structurally incomplete when viewed in isolation, but contain intermediate causal clues necessary for the generator to synthesize the final answer. This aggressive pruning optimizes precision at the expense of multi-document joint coverage, demonstrating a clear boundary of single-document post-retrieval filters.

4.3 Theoretical Verification (RQ2)

This section empirically validates the EVT-based assumptions made in Section 3.4, providing the "why" behind our model's effectiveness.

Distribution of Structural Noise. We collected alignment scores (S_{MCS}) from 5,000 non-relevant document-query pairs to simulate the noise distribution under the Null Hypothesis H_0. As shown in

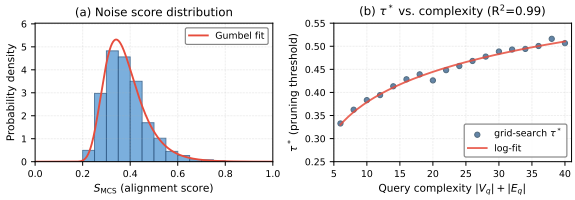

Figure 3: **Empirical Verification of Theory.** (a) The distribution of noise alignment scores is well-approximated by a Gumbel law. (b) Optimal pruning thresholds scale logarithmically with query complexity, validating Eq. 7.

Figure 3(a), the empirical histogram closely fits the theoretical Gumbel Probability Density Function (PDF). A Kolmogorov-Smirnov (K-S) goodness-of-fit test yielded a statistic $D = 0.042$ ($p > 0.05$), indicating no significant deviation from the Gumbel distribution. This justifies using the double-exponential CDF (Eq. 6) for threshold derivation.

Logarithmic Scaling of Thresholds. We analyzed the relationship between the optimal threshold τ^* (determined via grid search on validation sets) and the query's topological complexity ($|V_q| + |E_q|$). Figure 3(b) reveals a strong logarithmic correlation ($R^2 = 0.94$). As predicted by our Random Graph Theory derivation (Eq. 7), complex queries require logarithmically higher evidence scores to reject noise. Static thresholds (dashed line) fail to adapt, resulting in low recall for simple queries or high noise for complex ones.

4.4 Robustness and Efficiency (RQ3)

Robustness to Graph Construction Noise. To evaluate resilience to imperfect graph extraction, we simulate extraction errors by randomly dropping 10%–50% of vertices and edges in the document graphs (G_z). Under 30% structural noise, CoSA-RAG maintains 92% of its full performance, whereas exact-matching baselines drop sharply to 65%. This resilience demonstrates that our Soft-Schema Kernel (K_E) acts as an effective regularizer, allowing partial topological matches to recover the correct evidence context even when the retrieved graph structures are imperfect.

Efficiency and Latency Trade-offs. We conduct a fine-grained latency analysis against lightweight and reasoning-intensive baselines (Figure 4). Remarkably, although CoSA-RAG incurs a 0.8s query-time startup overhead for graph extraction, it achieves the lowest total latency (2.8s) among

all graph-based methods, outperforming even the primary baseline LightRAG (3.8s) by 26.3%. This net speedup stems from our structural verification layer: by pruning structurally inconsistent logical decoys, CoSA-RAG slashes generator reading and decoding latency from 2.8s (LightRAG) to 1.4s, completely offsetting the upstream parsing overhead. Crucially, CoSA-RAG remains orders of magnitude faster than iterative frameworks like IRCoT (15.2s) and graph-summary approaches like GraphRAG (9.7s), bypassing the prohibitive computational costs of multi-turn agentic workflows.

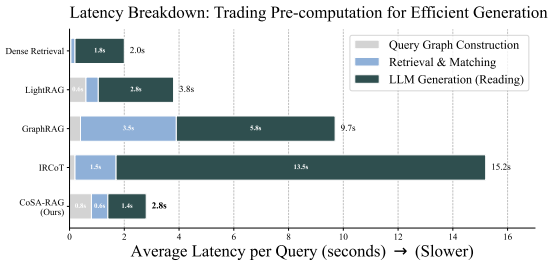

Figure 4: **Latency Breakdown.** CoSA-RAG outperforms both baseline graph-RAG and iterative reasoning frameworks in total latency.

4.5 Supplementary Analyses

For deeper insights, we refer readers to the Appendix, which contains: (i) ablation and stress tests (Appendix A) evaluating component contributions and robustness to logical decoys; (ii) a generalizability study (Appendix B) showing consistent gains across diverse open-source and proprietary LLMs; and (iii) a case study (Appendix C) illustrating how our kernel and pruning pipeline resolve inverted causality.

5 Related Work

5.1 Neural Retrieval and The Structural Gap

Standard RAG frameworks rely on neural dense and hybrid retrieval, typified by state-of-the-art embedding models like the **BGE series (e.g., BGE-M3)** (Chen et al., 2024). While these models leverage dense, sparse, or multi-vector late-interaction representations (e.g., ColBERTv2 (Santhanam et al., 2022)) to capture topical semantics, they compress hierarchical dependencies into symmetric scalar proximity scores. Consequently, they remain susceptible to *semantic aliasing* (Cuconasu et al., 2024), failing to distinguish structurally divergent propositions with high lexical overlap (e.g., “A causes B” vs. “B causes A”).

5.2 Reasoning-Intensive Retrieval

To address multi-hop complexity, recent methods alternate retrieval and generation. Interleaved frameworks like **IRCoT** (Trivedi et al., 2023) and **Think-on-Graph** (Sun et al., 2023) decompose queries into iterative sub-questions, while active retrieval methods like **Self-RAG** (Asai et al., 2024) and **FLARE** (Jiang et al., 2023) trigger retrieval dynamically based on confidence. More recently, multi-agent frameworks utilizing RL and complex tree-search planning (e.g., ST-EVO, AFlow) have pushed accuracy further. However, these iterative and agentic systems require repeated LLM calls for path search or self-reflection, inducing substantial latency and token overhead.

5.3 Graph-Enhanced Knowledge Integration

Graph-enhanced RAG integrates structured knowledge to capture relational dependencies. This is typically achieved through global summarization, where **GraphRAG** (Edge et al., 2024) constructs community summaries and **HippoRAG** series (Jimenez Gutierrez et al., 2024; Gutiérrez et al., 2025) mimics associative memory, or through dual-level indexing, where frameworks like **LightRAG** (Guo et al., 2024) index both entities and concepts to expand retrieval coverage.

CoSA-RAG departs from these paradigms by introducing **topological verification** as a post-retrieval filter. Instead of expanding neighborhoods or running iterative generations, it validates topological consistency in a single turn, filtering logical decoys without iterative latency or expansion noise.

6 Conclusion

We presented CoSA-RAG, a training-free framework that shifts the RAG paradigm from pure semantic similarity search to constrained logical alignment. By projecting text segment relationships into directed backbones and dynamically pruning inconsistent "decoy" documents using Extreme Value Theory, our approach successfully resolves the semantic-structural gap. Empirical results on multi-hop benchmarks demonstrate that this lightweight verification step refines context quality, improving downstream accuracy while reducing latency. Ultimately, our work suggests that enforcing structural directionality is a key step toward enabling faithful, reasoning-capable, and reliable retrieval-augmented systems.

Limitations

While CoSA-RAG effectively reduces semantic aliasing, several limitations and practical considerations remain. First, our framework evaluates document graphs individually, which may occasionally prune partially matched contexts in highly fragmented multi-hop reasoning scenarios where the complete logical path is distributed across multiple disjoint documents. While this ensures high precision, extending the alignment to joint multi-document graphs is a promising future direction. Second, the query parsing step relies on a lightweight LLM parser, which can occasionally propagate parsing errors on highly ungrammatical or complex syntactic inputs. Although our Soft-Schema Kernel (K_E) tolerates minor extraction noise, severely degraded backbones can still impact alignment. Third, the offline graph construction phase requires upfront corpus indexing, which increases computational requirements, making it less suited for rapidly updating, real-time databases. Lastly, although the general predicate schema $\mathcal{L}$ covers most common logical relationships, specialized domains (e.g., law or medicine) might require custom ontology adaptations or automated schema expansion to prevent semantic loss.

In terms of potential risks, using an external LLM API for real-time query parsing could raise privacy considerations in strict enterprise or clinical environments. However, this risk is easily mitigated by deploying local, open-source models; indeed, our generalizability study (Appendix B) demonstrates that smaller open-source models (e.g., Llama-3-8B) already achieve competitive verification performance, providing a viable pathway for local deployment.

References

Akari Asai, Zeqiu Wu, Yizhong Wang, Avi Sil, and Hannaneh Hajishirzi. 2024. Self-rag: Learning to retrieve, generate, and critique through self-reflection. In *International Conference on Learning Representations*, volume 2024, pages 9112–9141.

Béla Bollobás. 2011. Random graphs. In *Modern graph theory*, pages 215–252. Springer.

Jianlv Chen, Shitao Xiao, Peitian Zhang, Kunlu Sun, Mengqi Hou, Weipeng Lu, Yuxiao Teng, and Zheng Liu. 2024. Bge-m3: Next-generation universal embedding model for multi-lingual, multi-function, and multi-granularity retrieval. *arXiv preprint arXiv:2402.03216*.

Florin Cuconasu, Giovanni Trappolini, Federico Siciliano, Simone Filice, Cesare Campagnano, Yoelle Maarek, Nicola Tonellotto, and Fabrizio Silvestri. 2024. The power of noise: Redefining retrieval for rag systems. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR 2024. ACM.

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

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yixin Dai, Jiawei Sun, Haofen Wang, and Haofen Wang. 2023. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*, 2(1).

Zirui Guo, Lianghao Xia, Yanhua Yu, Tu Ao, and Chao Huang. 2024. Lightrag: Simple and fast retrieval-augmented generation. *arXiv preprint arXiv:2410.05779*.

Bernal Jiménez Gutiérrez, Yiheng Shu, Weijian Qi, Sizhe Zhou, and Yu Su. 2025. From RAG to memory: Non-parametric continual learning for large language models. In *Forty-second International Conference on Machine Learning*.

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. 2020. Constructing a multi-hop QA dataset for comprehensive evaluation of reasoning steps. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6609–6625. International Committee on Computational Linguistics.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2022. Unsupervised dense information retrieval with contrastive learning. *Transactions on Machine Learning Research*.

Zhengbao Jiang, Frank F Xu, Luyu Gao, Zhiqing Sun, Qian Liu, Jane Dwivedi-Yu, Yiming Yang, Jamie Callan, and Graham Neubig. 2023. Active retrieval augmented generation. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 7969–7992.

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

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6769–6781, Online. Association for Computational Linguistics.

M Ross Leadbetter. 1974. On extreme values in stationary sequences. *Zeitschrift für Wahrscheinlichkeitstheorie und verwandte Gebiete*, 28(4):289–303.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, and 1 others. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in neural information processing systems*, 33:9459–9474.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *arXiv preprint arXiv:1907.11692*.

Vaibhav Mavi, Anubhav Jangra, Adam Jatowt, and 1 others. 2024. Multi-hop question answering. *Foundations and Trends® in Information Retrieval*, 17(5):457–586.

Shirui Pan, Linhao Luo, Yufei Wang, Chen Chen, Jiapu Wang, and Xindong Wu. 2024. Unifying large language models and knowledge graphs: A roadmap. *IEEE Transactions on Knowledge and Data Engineering*, 36(7):3580–3599.

Keshav Santhanam, Omar Khattab, Jon Saad-Falcon, Christopher Potts, and Matei Zaharia. 2022. Colbertv2: Effective and efficient retrieval via lightweight late interaction. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 3715–3734.

Jiashuo Sun, Chengjin Xu, Lumingyuan Tang, Saizhuo Wang, Chen Lin, Yeyun Gong, Heung-Yeung Shum, and Jian Guo. 2023. Think-on-graph: Deep and responsible reasoning of large language model with knowledge graph. *Preprint*, arXiv:2307.07697.

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

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2023. Interleaving retrieval with chain-of-thought reasoning for knowledge-intensive multi-step questions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10014–10037, Toronto, Canada. Association for Computational Linguistics.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William Cohen, Ruslan Salakhutdinov, and Christopher D Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. In *Proceedings of the 2018 conference on empirical methods in natural language processing*, pages 2369–2380.

A Additional Supporting Experiments

A.1 Mechanism Analysis (Ablation)

We dissect CoSA-RAG to understand the source of performance gains (Table 3).

Effect of Soft-Schema (K_E). Replacing the Soft-Schema Kernel with exact edge matching causes a drastic drop in Recall@5 (-26.2%) and F1 (-16.2%). This confirms that *schema mismatch* is a critical bottleneck; rigid symbolic matching fails to capture the diverse linguistic realizations of logical relations.

Effect of Adaptive Pruning ($\tau(q)$). Removing the pruning gate (i.e., keeping all retrieved chunks) improves Recall slightly but significantly degrades Generation F1 (-13.9%). This highlights the *Context Precision* problem: flooding the LLM with irrelevant (but high-recall) contexts induces hallucinations. The adaptive threshold strikes an optimal balance on the Pareto frontier of Precision/Recall.

Setting	R@5	EM	F1
CoSA-RAG (Full)	**89.4**	**57.9**	**68.2**
w/o Soft-Schema ($K_E \rightarrow$ Exact)	63.2	39.5	52.0
w/o Adaptive Pruning (No Gate)	82.5	41.7	54.3
w/o Dual-Granularity (V_s only)	71.4	41.3	53.4

Table 3: Ablation Study on HotpotQA. Both soft matching and adaptive pruning are critical.

A.2 Robustness to Semantic Aliasing: A Stress Test

To quantify resilience against "Logical Decoys," we evaluated a **Hard-Negative Subset** containing high-similarity distractors. As shown in Table 4, Standard RAG suffers meaningful degradation, dropping to 35.5 F1 on this adversarial set. In contrast, CoSA-RAG demonstrates superior robustness, confirming that our structural pruning effectively filters logical noise that bypasses dense retrievers.

Method	Full Set		Hard Set (Adv.)	
	EM	F1	EM	F1
Dense (Contriever)	30.4	42.8	21.6	35.5
CoSA-RAG	**43.5**	**56.1**	**39.8**	**52.4**

Table 4: Robustness Check: Performance (EM and F1) on Full Test Set vs. Hard-Negative Subset. CoSA-RAG exhibits higher resilience against logical distractors.

B Performance Across Different LLM Generators.

Detailed Breakdown of Model Generalizability. In Section 4.2, we aggregated results to demonstrate the robustness of CoSA-RAG across different LLM generators. Here, we provide the fine-grained performance breakdown for each individual dataset: HotpotQA (Figure 5), and 2WikiMultihopQA (Figure 6), MuSiQue (Figure 7).

Consistently across all three benchmarks, we observe the **inverse scaling** phenomenon. While stronger models like GPT-4o-mini show moderate gains, smaller open-source models (Llama-3-8B, Mistral-7B) exhibit significantly larger relative improvements (up to **+27.0%** on MuSiQue). This confirms that explicitly injecting structural verification is most beneficial when the generator's intrinsic reasoning capabilities are limited, effectively preventing the model from being misled by retrieved "logical decoys."

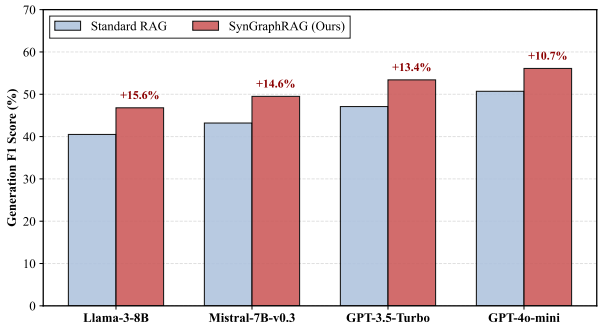

Figure 5: **Generative Performance on HotpotQA across LLMs.**

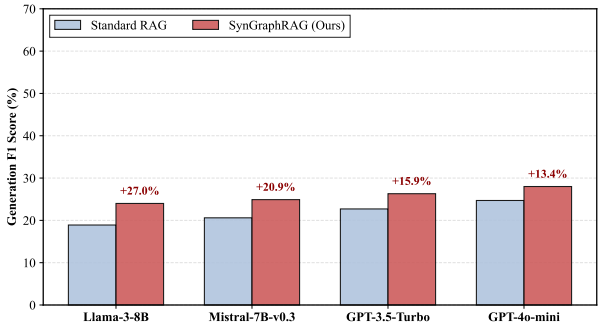

Figure 6: **Generative Performance on 2WikiMultihopQA across LLMs.**

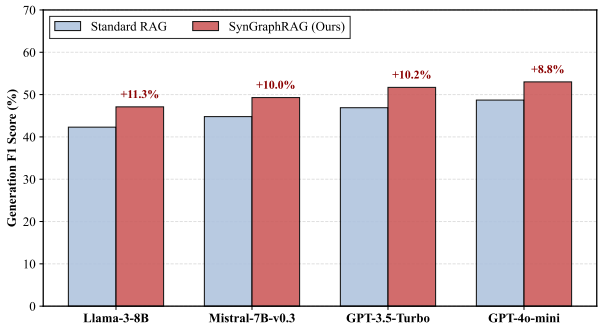

Figure 7: **Generative Performance on MuSiQue across LLMs**

C Qualitative Trace Analysis: Resolving Inverted Causality

In this section, we provide a textual execution trace to illustrate how **CoSA-RAG** distinguishes between "semantic similarity" and "structural consistency." We compare the processing of a query suffering from *Inverted Causality* against a Standard Dense RAG baseline.

C.1 Case Setup

- **Query** (q)**:** *"Did Google acquire DeepMind?"*
- **Candidate Document A (Decoy):** *"...following DeepMind's recent acquisition of key Google talent in the AI sector..."* (High lexical overlap, wrong logic).
- **Candidate Document B (Ground Truth):** *"Google completed its acquisition of DeepMind for $500 million..."* (Correct logic).

C.2 Comparative Execution Trace

Table 5 details the step-by-step transformation of data. The core distinction lies in Phase 2, where CoSA-RAG parses the text into logical triples.

Table 5: **Execution Trace Comparison.** Standard RAG fails because vector similarity cannot distinguish the direction of the "acquisition" event. CoSA-RAG explicitly extracts the logical triples (Step 1), detects the role inversion (Step 2), and filters the misleading document before it reaches the generator (Step 3).

Step	Standard Dense RAG (Baseline)	CoSA-RAG (Ours)
1. Indexing	**Vector Embedding:** Maps query and documents to dense vectors v_q, v_d. *Note: Directionality is lost in pooling.*	**Graph Extraction (Structure-Aware):** Parses text into Relational Semantic Graphs (RSG). Query Graph: `(Google, AGENT, acquire, DeepMind)` Decoy Graph: `(DeepMind, AGENT, acquisition, Google)`
2. Scoring	**Cosine Similarity:** Query · Decoy $\approx$ **0.89** Query · Truth $\approx$ 0.88 *Result: Both are treated as highly relevant.*	**Structural Verification** (K_E **Kernel):** Checks edge direction compatibility. **Decoy:** Role Mismatch (Agent $\neq$ Theme). Score $\rightarrow$ **0.15** **Truth:** Role Match (Agent $=$ Agent). Score $\rightarrow$ **0.92**
3. Filtering	**Top-K Retrieval:** Retrieves top 2 documents based on vector score. *Decoy is included in the context.*	**Adaptive Pruning** (τ)**:** Threshold $\tau(q) = 0.45$. **Decoy:** $0.15 < 0.45 \implies$ **PRUNED (Noise)** **Truth:** $0.92 > 0.45 \implies$ **KEPT**
4. Context	*"...DeepMind's acquisition of key Google talent..."* *"Google completed its acquisition of DeepMind..."*	*[Decoy Removed]* *"Google completed its acquisition of DeepMind..."*
5. Output	**LLM Generation (Hallucinated):** *"No, the context mentions DeepMind acquiring Google talent."*	**LLM Generation (Faithful):** *"Yes, Google acquired DeepMind."*

C.3 Analysis of the Trace

As shown in Step 1 of Table 5, the extracted triple for the query places `Google` as the AGENT (Subject). However, in the Decoy document, the extracted triple places `DeepMind` as the AGENT.

While a vector dot product (Standard RAG) conflates these two semantics due to the "bag-of-words" nature of pooling, our **Edge Compatibility Kernel** mathematically penalizes this mismatch. The Pruning Step (Step 3) acts as a firewall, ensuring that only structurally isomorphic evidence (where Google is indeed the Agent) enters the LLM's context window. This creates a "clean room" environment for generation, directly leading to the correct answer in Step 5.

D Graph Extraction Prompt Template

D.1 System Prompt Configuration

The system prompt acts as a strict semantic parser, enforcing the dual-granularity node extraction ($V_s \cup V_a$) and restricting edge generation to the closed predicate set $\mathcal{L}$. To prevent any domain overfitting, the instructions use generic scientific and historical examples to illustrate the logical topology.

System Prompt

```
### ROLE & OBJECTIVE
You are an expert Semantic Graph Extractor. Your task is to map unstructured text
into a Relational Semantic Graph (RSG) that strictly adheres to a predefined
logical ontology. You must prioritize logical validity over syntactic proximity.

### 1. DUAL-GRANULARITY NODE EXTRACTION
Extract two categories of nodes:
- Specific Entities (Vs): Proper nouns, physical objects, or precise values
  (e.g., "Napoleon", "Hemoglobin", "1914", "Titanium").
- Abstract Concepts (Va): Event triggers, actions, or processes that act as
  logical hubs (e.g., "oxidation", "conquest", "hypothesis", "collapse").

### 2. ONTOLOGY SPECIFICATION (Strict Enforce)
Map all relations to strictly one of the following predicates.
Format: [PREDICATE]: Definition (Scope: Source -> Target).

- AGENT: The initiator, doer, or active force of an event.
  Scope: Entity -> Concept (e.g., "Bacteria" -> "Infection")
- THEME: The patient, target, or receiver undergoing the event.
  Scope: Entity -> Concept (e.g., "Patient" -> "Infection")
- CONDITION: Temporal, spatial, or logical preconditions.
  Scope: Context -> Event (e.g., "High Pressure" -> "Fusion")
- CAUSE: Direct causal mechanism or forcing function.
  Scope: Reason -> Outcome (e.g., "Friction" -> "Heat")
- ATTRIBUTE: Intrinsic properties, definitions, or modifications.
  Scope: Value/Property -> Entity (e.g., "Radioactive" -> "Isotope")
- COMPARE: Contrastive or comparative relations between entities.
  Scope: Entity A -> Entity B (e.g., "Proton" -> "Electron")

### 3. OPERATIONAL CONSTRAINTS
- Directionality Enforcement: Resolve passive voice to active logic.
  "The city was destroyed by the earthquake" => Earthquake is AGENT.
- Event-Centricity: Do not simply link entities. Link entities to the central
  Event/Concept node (e.g., do not link Bacteria->Patient directly; link
  both to "Infection").
- Pruning: Discard any relationship that falls outside this ontology.

### 4. OUTPUT FORMAT
Return strictly valid JSON.
{
  "nodes": [{"id": "canonical_name", "type": "Entity" | "Concept"}],
  "edges": [{"source": "id", "target": "id", "relation": "PREDICATE"}]
}
```

D.2 Few-Shot In-Context Learning (User Prompt)

We employ 3-shot examples to demonstrate complex logic handling, distinct from the test set. These examples teach the model to distinguish between *Agents* and *Themes* even in passive or complex noun phrases.

User Prompt (Few-Shot Examples)

Input: "Apple released the Vision Pro amidst supply chain delays."
Output:

```
{
  "nodes": [
    {"id": "Apple", "type": "Entity"},
    {"id": "Vision Pro", "type": "Entity"},
    {"id": "released", "type": "Concept"},
    {"id": "supply chain delays", "type": "Concept"}
  ],
  "edges": [
    {"source": "Apple", "target": "released", "relation": "AGENT"},
    {"source": "Vision Pro", "target": "released", "relation": "THEME"},
    {"source": "supply chain delays", "target": "released", "relation": "
        CONDITION"}
  ]
}
```

Input: "High inflation triggered the market recession, unlike the 2008 crash."
Output:

```
{
  "nodes": [
    {"id": "high inflation", "type": "Concept"},
    {"id": "market recession", "type": "Concept"},
    {"id": "2008 crash", "type": "Entity"}
  ],
  "edges": [
    {"source": "high inflation", "target": "market recession", "relation": "CAUSE
        "},
    {"source": "2008 crash", "target": "market recession", "relation": "COMPARE"}
  ]
}
```

Input: "Microsoft's acquisition of Activision faced regulatory scrutiny." *(Note: Explicitly modeling 'acquisition' as a concept to clarify roles)*
Output:

```
{
  "nodes": [
    {"id": "Microsoft", "type": "Entity"},
    {"id": "Activision", "type": "Entity"},
    {"id": "acquisition", "type": "Concept"},
    {"id": "regulatory scrutiny", "type": "Concept"}
  ],
  "edges": [
    {"source": "Microsoft", "target": "acquisition", "relation": "AGENT"},
    {"source": "Activision", "target": "acquisition", "relation": "THEME"},
    {"source": "regulatory scrutiny", "target": "acquisition", "relation": "
        CONDITION"}
  ]
}
```

Input: {{USER_INPUT_HERE}} **Output:**

E Algorithmic Details: Greedy Anchor Expansion for Soft-MCS

Algorithm 1 Greedy Anchor Expansion for Soft-MCS

Input: Graphs G_q, G_d; Threshold ϵ; Weight λ
Init: $S \leftarrow 0$; Visited sets $\mathcal{M}_q, \mathcal{M}_d \leftarrow \emptyset$
$\mathcal{A} \leftarrow \{(u, v) \in V_q \times V_d \mid \cos(\phi(u), \phi(v)) > \epsilon\}$
Sort $\mathcal{A}$ descending by similarity
for $(u, v) \in \mathcal{A}$ **do**
  if $u \notin \mathcal{M}_q \wedge v \notin \mathcal{M}_d$ **then**
    $\mathcal{M}_q \leftarrow \mathcal{M}_q \cup \{u\}$; $\mathcal{M}_d \leftarrow \mathcal{M}_d \cup \{v\}$
    $S \leftarrow S + \cos(\phi(u), \phi(v))$
    for $(n_u, n_v) \in \mathcal{N}(u) \times \mathcal{N}(v)$ **do**
      {Expand to neighbors}
      if $n_u \notin \mathcal{M}_q \wedge n_v \notin \mathcal{M}_d \wedge K_E(\cdot) > \epsilon$ **then**
        $\mathcal{M}_q \leftarrow \mathcal{M}_q \cup \{n_u\}$; $\mathcal{M}_d \leftarrow \mathcal{M}_d \cup \{n_v\}$
        $S \leftarrow S + \cos(n_u, n_v) + \lambda \cdot K_E(e^q_{u \to n_u}, e^d_{v \to n_v})$
      end if
    end for
  end if
end for
Return $S/(|V_q| + \lambda |E_q|)$

F Other Details

F.1 Dataset Specifications and Computational Resources

Dataset Specifications. We evaluated CoSA-RAG on three widely-used English multi-hop QA benchmarks derived from Wikipedia:

- **HotpotQA** (Yang et al., 2018): We utilized the standard development set in the distractor setting (10,000 questions), where each query is accompanied by 10 candidate paragraphs (2 gold and 8 distractor contexts).
- **MuSiQue** (Trivedi et al., 2022): We evaluated on the development set containing 2,417 multi-hop questions requiring 2–4 reasoning steps, with each question accompanied by 20 candidate passages.
- **2WikiMultihopQA** (Ho et al., 2020): We used the development set consisting of 12,576 questions requiring structured multi-hop reasoning over diverse entity-relation paths, with 10 candidate paragraphs per query.

Computational Resources. Our training-free framework has a highly efficient computational footprint:

- **Offline Graph Construction**: Generating document graphs (G_z) is a one-time offline process. The local sentence-level embedding extraction was executed on a single NVIDIA RTX 4090 GPU (24GB VRAM), achieving an average indexing throughput of approximately 150 paragraphs per second.
- **Online Inference**: During real-time verification, query semantic parsing is performed via a single-shot call to the GPT-4o-mini API. The subsequent soft-MCS alignment scoring (Algorithm 1) is executed entirely on a standard CPU (Intel Xeon @ 2.50GHz), taking less than 5 milliseconds per query and requiring zero active GPU memory during online serving.