DHIRAG: LEARNING SOFT HIERARCHICAL INDICES FOR RETRIEVAL-AUGMENTED GENERATION

Anonymous authors
Paper under double-blind review

ABSTRACT

Hierarchical indices can provide corpus-level context for retrieval-augmented generation, but their usefulness depends on both how communities are constructed and how community information is combined with passage-level relevance. We present DhiRAG, a framework that learns soft hierarchical affiliations and uses them to contextualize evidence ranking. Starting from a semantic graph over text chunks, a graph neural assignment network is optimized using a differentiable structural-entropy objective. The learned affiliations serve two connected roles: they produce community prototypes through weighted aggregation of chunk embeddings, and they transfer query–community relevance to the original chunks. The resulting structural scores are combined with local matching signals to retrieve textual evidence, without generating natural-language community summaries. On PopQA, HotpotQA, 2WikiMultiHopQA, and MuSiQue, DhiRAG achieves the highest average Str-Acc and LLM-Acc among the evaluated methods, improving over the strongest average baseline by 2.4 and 2.7 percentage points, respectively. Retrieval-level evaluation and component ablations provide complementary evidence for the effectiveness of the learned index. These results support soft hierarchical context as a useful complement to local evidence matching.

1 INTRODUCTION

Retrieval-augmented generation (RAG) augments large language models (LLMs) with evidence from an external corpus (Lewis et al., 2020). For questions that require information from multiple passages, effective retrieval involves more than identifying text that is individually similar to the query. A passage containing an intermediate fact may become useful only when considered alongside other evidence. This motivates combining direct query–passage matching with corpus-level context that captures relationships among passages. Graph-based and hierarchical retrieval offer complementary ways to incorporate such context into evidence selection (Jimenez Gutierrez et al., 2024; Sarthi et al., 2024; Edge et al., 2024).

A central design question is how to construct this organization and make it useful for passage-level retrieval. Existing approaches represent related passages through graph connections, recursive textual summaries, or hierarchical vector representations (Jimenez Gutierrez et al., 2024; Sarthi et al., 2024; Edge et al., 2024; Wei et al., 2026). These representations provide information beyond individual passages, but their usefulness depends on how they are connected to evidence ranking. In particular, a passage may relate to several semantic contexts, and the context most relevant to retrieval can vary with the query. Treating community membership solely as an exclusive routing decision can limit how this contextual information is used. We instead investigate a hierarchy whose affiliations serve both to represent the corpus and to inform the ranking of its passages.

We propose **DhiRAG**, a framework that connects learned hierarchical indexing and evidence retrieval through shared soft affiliations. DhiRAG constructs a semantic KNN graph over chunk embeddings and uses a graph neural assignment network to predict parent affiliations at successive levels. Building on differentiable structural-information formulations for graph clustering (Sun et al., 2024; Yu et al., 2025), the network is optimized with a continuous structural-entropy objective. This objective provides a graph-based learning criterion for hierarchical organization, while message passing allows the assignments to depend on both chunk representations and their neighborhood

Reviewers: please read the Reviewer Guidelines (iclr.cc/Conferences/2027/ReviewerGuidelines) and the AI Policy for Reviewers (iclr.cc/Conferences/2027/AIPolicyForReviewers). If you used AI to expand, edit, or polish your review, please provide the input text to the LLM. Better still, consider skipping the LLM and submitting your original text: we, and the authors, are much more interested in your unedited thoughts than in what an LLM has to say. AI-assisted or not, you are putting your name and reputation behind your review: LLM-generated falsehoods, hallucinations or misrepresentations are subject to disciplinary action, which may include desk-rejecting all papers you have authored.

context. The resulting affiliations express graded relationships between chunks and communities rather than requiring each chunk to contribute to only one community.

The same affiliations connect index construction to query-time scoring. During indexing, cumulative chunk-to-community weights aggregate the original chunk embeddings into community prototypes. These prototypes provide vector-space abstractions without requiring LLM-generated community summaries. During retrieval, query–community similarities are projected back to chunks through their learned affiliations, producing a structural relevance signal. DhiRAG combines this signal with local semantic and lexical matching in a single ranking stage. The hierarchy therefore supplies context for evidence selection rather than determining candidate eligibility through successive community-level filtering decisions. The final reader receives the retrieved original passages, while community prototypes remain internal representations of the retriever.

We evaluate DhiRAG on PopQA, HotpotQA, 2WikiMultiHopQA, and MuSiQue, covering factual and multi-hop question answering. DhiRAG achieves the highest average performance under both String-based Accuracy and LLM-based Accuracy among the evaluated methods, with dataset-dependent gains. Direct retrieval measurements, component ablations, and sensitivity analyses complement the end-to-end evaluation, while indexing-cost measurements characterize the computational trade-offs of learning and using the hierarchy.

Our contributions are threefold:

- We develop a learned hierarchical indexing framework that combines graph message passing with differentiable structural entropy to infer non-exclusive chunk-to-community affiliations.
- We use a shared affiliation representation for both summary-free community construction and community-informed passage scoring, integrating hierarchical context with local retrieval signals.
- We evaluate the framework across factual and multi-hop QA benchmarks, with direct retrieval measurements, ablations, sensitivity analyses, and indexing-cost comparisons that characterize its effectiveness and computational trade-offs.

2 Related Work

Graph-Based RAG. Graph-based retrieval augments passage-level matching with relationships represented in a graph. LightRAG (Guo et al., 2025) combines graph structure with vector representations to retrieve information associated with entities and their relationships. HippoRAG (Jimenez Gutierrez et al., 2024) and HippoRAG2 (Gutiérrez et al., 2025) use Personalized PageRank (Haveliwala, 2002) to support associative retrieval over graph-structured knowledge. Other approaches learn or optimize the retrieval of graph context: G-Retriever (He et al., 2024) selects relevant subgraphs for question answering over textual graphs, while GFM-RAG (Luo et al., 2026) develops a graph neural retriever that transfers across knowledge graphs. These methods illustrate different ways of exploiting graph connectivity beyond independent passage similarity. DhiRAG studies a complementary design in which text chunks are graph nodes and learned hierarchical affiliations provide contextual signals for ranking the original chunks.

Hierarchical RAG. Hierarchical indices represent a corpus at multiple levels of abstraction. RAPTOR (Sarthi et al., 2024) recursively combines soft clustering with textual summarization and supports both tree traversal and collapsed-tree retrieval. CAM (Li et al., 2025) uses incremental overlapping clustering to organize hierarchical memory. MS GraphRAG (Edge et al., 2024) constructs community summaries for query-focused synthesis, while ArchRAG (Wang et al., 2026a) develops an attributed-community hierarchy for retrieval. E^2GraphRAG (Zhao et al., 2025) combines a summary tree with an entity graph to support local and global retrieval. Alternatively, SeRAG (Wei et al., 2026) uses structural-entropy indexing and vector-based community representations without LLM-generated community summaries. These approaches differ in how they construct communities, represent higher-level information, and connect the hierarchy to retrieval. DhiRAG shares the goal of summary-free hierarchical indexing, but learns soft affiliations through a graph neural network and reuses their cumulative weights in both community-prototype construction and chunk-level scoring.

Structural Information and Learned Hierarchies. Structural information theory provides an information-theoretic criterion for organizing a graph through a partitioning tree (Li & Pan, 2016; Li, 2024). LSEnet (Sun et al., 2024) introduces differentiable structural information for neural graph clustering, and HyperSED (Yu et al., 2025) applies related ideas to unsupervised social event detection. DhiRAG builds on these formulations to learn a corpus-level hierarchical index. Its focus is the use of the learned affiliations as a shared representation for community construction and query-time evidence ranking, rather than a new structural-information theory or a GNN architecture in isolation.

3 PRELIMINARY

3.1 CORPUS GRAPHS AND HIERARCHICAL AFFILIATIONS

We represent a corpus as a collection of n text chunks, $\mathcal{V} = \{v_1, v_2, \ldots, v_n\}$. Their relationships are modeled by an undirected graph $G = (\mathcal{V}, \mathcal{E}, W)$, where $W \in \mathbb{R}_{\geq 0}^{n \times n}$ is a symmetric adjacency matrix with $w_{ii} = 0$. In DhiRAG, the graph is induced by similarities between chunk embeddings; it does not assume the availability of an explicit entity–relation knowledge graph or encode every document-native relationship. The weighted degree of node v_i is $d_i = \sum_j w_{ij}$, and the graph volume is $\mathrm{Vol}(G) = \sum_i d_i$. Throughout the matrix-based expressions below, a sum over $\mathcal{E}$ enumerates both ordered adjacency entries of each undirected edge.

A hierarchical index groups chunks into progressively coarser communities. In a discrete partitioning tree, each chunk corresponds to a leaf, and each non-root tree node has a unique parent. DhiRAG relaxes this exclusive parent assignment into graded affiliations, allowing a chunk to contribute to more than one community representation. These affiliations are used to construct corpus-level context and to inform passage ranking. They are representation weights rather than annotations of independently identified semantic topics.

3.2 DISCRETE STRUCTURAL INFORMATION

Structural information theory characterizes graph organization through a partitioning tree $\mathcal{T}$ (Li & Pan, 2016; Li, 2024). Let λ denote its root. Every graph node corresponds to a leaf of $\mathcal{T}$. For a tree node α, let $T_\alpha \subseteq \mathcal{V}$ be the set of graph nodes represented by its descendant leaves. Thus, $T_\lambda = \mathcal{V}$, and the children of an internal node partition its descendant set in the discrete formulation. We write $\alpha^- = \mathrm{parent}_{\mathcal{T}}(\alpha)$ for the parent of α in the tree; this relation does not refer to an edge direction in the undirected graph.

Definition 1 (H-dimensional Structural Entropy). Given a graph $G = (\mathcal{V}, \mathcal{E}, W)$ with $\mathrm{Vol}(G) > 0$ and a partitioning tree $\mathcal{T}$ of height H, its structural entropy with respect to $\mathcal{T}$ is defined as:

$$\mathcal{H}^{\mathcal{T}}(G) = \sum_{\alpha \in \mathcal{T},\, \alpha \neq \lambda} \mathcal{H}^{\mathcal{T}}(G; \alpha) = - \sum_{\alpha \in \mathcal{T},\, \alpha \neq \lambda} \frac{g_\alpha}{\mathrm{Vol}(G)} \log_2 \frac{V_\alpha}{V_{\alpha^-}}, \tag{1}$$

where $V_\alpha = \sum_{v_i \in T_\alpha} d_i$ is the volume of the descendant set of α, and V_{α^-} is the volume of its parent module. The boundary weight is $g_\alpha = \sum_{i:\, v_i \in T_\alpha} \sum_{j:\, v_j \notin T_\alpha} w_{ij}$: each edge with one endpoint inside T_α and the other outside contributes once. Zero-volume modules contribute zero under the usual limiting convention.

Equation equation 1 assigns a structural cost to a candidate hierarchical partition. Minimizing this cost over an admissible family of trees defines an information-theoretic graph-organization problem. However, the discrete assignment decisions are not directly amenable to gradient-based learning. Following differentiable structural-information formulations (Sun et al., 2024; Yu et al., 2025), Section 4 parameterizes parent affiliations with a neural network. The resulting continuous objective is used to learn the index; its usefulness for retrieval is evaluated empirically.

4 METHODOLOGY

DhiRAG connects hierarchical index learning and evidence ranking through shared soft affiliations. The framework consists of three stages: (i) constructing a semantic graph over text chunks; (ii)

learning hierarchical affiliations and using them to form community prototypes; and (iii) combining affiliation-weighted community relevance with local passage matching. The hierarchy supplies contextual information for ranking, while the final evidence passed to the reader consists of original text chunks.

4.1 Semantic Graph Construction

To model the multifaceted dependencies between information units, we transform the raw text chunks $\mathcal{V} = \{v_1, v_2, \ldots, v_n\}$ into a weighted, undirected semantic graph $G = (\mathcal{V}, \mathcal{E}, W)$. We first encode each chunk v_i into a high-dimensional dense vector $\mathbf{x}_i \in \mathbb{R}^d$ using a pre-trained language model, yielding the feature matrix $\mathbf{X} \in \mathbb{R}^{n \times d}$.

We construct a K-nearest neighbor (KNN) graph where the edge weight w_{ij} between nodes v_i and v_j is defined by their normalized cosine similarity:

$$w_{ij} = \max(0, \cos(\mathbf{x}_i, \mathbf{x}_j)) \cdot \mathbb{I}(j \in \mathcal{N}_K(i) \vee i \in \mathcal{N}_K(j)), \tag{2}$$

where $\mathcal{N}_K(i)$ denotes the K-nearest neighbors of v_i. The resulting adjacency matrix $W \in \mathbb{R}^{n \times n}$ directly parameterizes the underlying topological density of the corpus. Crucially, W serves as the foundational numerical input for our subsequent structural entropy optimization, ensuring that the graph's edge weights accurately govern the expected boundary cuts during community discovery.

4.2 Differentiable Hierarchical Indexing

We learn a continuous hierarchical representation of the corpus by parameterizing non-exclusive parent affiliations. The learned weights connect two stages of the pipeline: they aggregate chunk embeddings into community prototypes during indexing and transfer community relevance to individual chunks during retrieval. The graph objective provides a corpus-level learning criterion, while the usefulness of the resulting index is assessed empirically.

Level-wise and Global Soft Assignments. We conceptualize the hierarchical organization through a level-wise soft assignment matrix $\mathbf{C}^h \in [0,1]^{N_h \times N_{h-1}}$ at each height h, where C^h_{ij} denotes the probability of the i-th node at level h belonging to the j-th parent community. To bridge the fundamental information units (leaf nodes) to any high-level abstract community, we define the global affiliation matrix $\mathbf{S}^h$ as the cumulative product of the level-wise assignments:

$$\mathbf{S}^h = \prod_{k=H+1}^{h+1} \mathbf{C}^k, \quad \mathbf{C}^{H+1} = \mathbf{I}_n. \tag{3}$$

The element $S^h_{ik} \in [0,1]$ encapsulates the soft semantic overlap, representing the precise probability that the raw chunk v_i acts as a constituent of community k.

Differentiable Structural-Entropy Objective.

Building on the level-wise formulation of structural information (Sun et al., 2024; Yu et al., 2025), we use soft affiliations to construct a differentiable graph-organization objective rather than heuristic clustering (Zhao et al., 2025; Li et al., 2025; Wang et al., 2026a; Zhang et al., 2026). First, the expected volume of community k at level h is computed by aggregating the node degrees d_i weighted by their global affiliation probabilities. Simultaneously, the expected volume of its parent node k^- is elegantly extracted by marginalizing over the level-wise assignments:

$$V^h_k = \sum_{i=1}^{n} S^h_{ik} d_i, \qquad V^{h-1}_{k^-} = \sum_{k'=1}^{N_{h-1}} \mathbf{C}^h_{kk'} V^{h-1}_{k'}. \tag{4}$$

The core mathematical bottleneck of classical structural entropy lies in computing the discrete boundary cut g_α. We bypass this by observing that the boundary cut equals the community's total expected volume minus its internal edge weights. Using $\mathbf{S}^h$ as a probabilistic mask, the differentiable expected cut g^h_k is derived as:

$$g^h_k = V^h_k - \sum_{(i,j) \in \mathcal{E}} S^h_{ik} S^h_{jk} w_{ij}. \tag{5}$$

In this formulation, the term $S_{ik}^h S_{jk}^h$ acts as a soft gating mechanism: an edge weight w_{ij} is fully retained as an internal edge only if both endpoints exhibit high probabilities of belonging to community k. By integrating these differentiable components, the overall continuous structural entropy $\mathcal{H}^{\mathcal{T}}(G)$ is formalized as:

$$\mathcal{H}^{\mathcal{T}}(G) = \sum_{h=1}^{H} \mathcal{H}^{\mathcal{T}}(G;h) = -\frac{1}{\text{Vol}(G)} \sum_{h=1}^{H} \sum_{k=1}^{N_h} \left(V_k^h - \sum_{(i,j)\in\mathcal{E}} S_{ik}^h S_{jk}^h w_{ij} \right) \log_2 \frac{V_k^h}{V_{k^-}^{h-1}}. \tag{6}$$

Neural Instantiation via SENet. While minimizing Eq. (6) could theoretically be achieved using a naive Multi-Layer Perceptron (MLP) to output assignment probabilities, such an approach would rely exclusively on topological connectivity, ignoring the rich textual semantics of the chunks. To enforce a strict synergy between structural hierarchy and semantic coherence, we instantiate this continuous optimization framework through a Structure Entropy Neural Network (SENet).

At each hierarchical level h, SENet employs a residual Graph Convolutional Network (GCN) (Kipf & Welling, 2017) to perform message passing. This operation updates the node representations $\mathbf{X}^h$ by aggregating contextual information from topological neighbors, effectively aligning the latent semantic features with the graph structure. A non-linear assigner then projects these structure-aware representations to compute the soft assignment matrix:

$$\mathbf{C}^h = \text{Gumbel-Softmax}(\text{MLP}(\text{GCN}(\mathbf{X}^h, \mathbf{A}^h)); \tau), \tag{7}$$

where τ is a temperature hyperparameter controlling the crispness of the assignments, and $\mathbf{A}^h$ is the adjacency matrix at level h.

To propagate the topological structure upward, we perform graph pooling by recursively computing the adjacency matrix for the next hierarchical level as $\mathbf{A}^{h-1} = (\mathbf{C}^h)^T \mathbf{A}^h \mathbf{C}^h$. The diagonal elements of this coarsened matrix $\mathbf{A}^{h-1}$ inherently represent the internal edge weights of the newly formed communities, which serves as the exact numerical foundation for computing g_k^h. To prevent community feature collapse, a degenerate solution where all nodes collapse into a singular community, we augment the structural entropy objective with an orthogonal regularization term. Consequently, the entire SENet is trained end-to-end by minimizing the combined loss function $\mathcal{L}$:

$$\mathcal{L} = \mathcal{H}^{\mathcal{T}}(G) + \lambda \sum_{h} \|(\mathbf{C}^h)^T \mathbf{C}^h - \mathbf{I}\|_F^2, \tag{8}$$

where λ is a weighting coefficient. By minimizing Eq. (8) via backpropagation, SENet guarantees that the learned hierarchical assignments $\mathbf{S}^h$ are jointly optimized for topological density and semantic similarity, laying a robust foundation for downstream representation.

Affiliation-Weighted Community Prototypes. Because the learned soft assignments $\mathbf{S}^h$ intrinsically capture both structural community boundaries and semantic coherence, we can synthesize high-level abstractions for each community directly in the latent vector space, bypassing the prohibitive token costs of LLM-based text generation. Specifically, the semantic centroid $\mathbf{E}_k$ for community k is computed as the probability-weighted aggregation of its constituent chunks' raw embeddings:

$$\mathbf{E}_k = \text{Normalize}\left(\sum_{i=1}^{n} S_{ik}^h \mathbf{x}_i \right). \tag{9}$$

Instead of forcing a multi-topic chunk into a single centroid, its unique semantic traits are proportionally distributed across all its affiliated community vectors based on the semantically-grounded probability S_{ik}^h.

4.3 Probability-Weighted Single-Pass Retrieval

Existing GraphRAG pipelines often employ a multi-stage, coarse-to-fine filtering paradigm. However, utilizing high-level abstractions for hard filtering prematurely truncates the search space, which contradicts our continuous probability modeling. To effectively leverage the learned soft affiliation index, DhiRAG abandons cascaded filtering in favor of a single-pass weighted fusion scoring mechanism.

Structural Relevance from Soft Affiliations. Given a user query q embedded as $\mathbf{x}_q$, we first evaluate how well each chunk aligns with the query from a macroscopic, structural perspective. We project the community-level relevance directly onto the chunks using the learned soft assignments $\mathbf{S}^h$. Because a chunk v_i belongs to multiple communities with varying probabilities, its structural prior score $\mathcal{S}_{prior}(q, v_i)$ is computed as the maximum probability-decayed community match:

$$\mathcal{S}_{prior}(q, v_i) = \max_k \Big(\cos(\mathbf{x}_q, \mathbf{E}_k) \cdot S^h_{ik} \Big). \tag{10}$$

This formulation acts as a dynamic structural weighting factor rather than a threshold. A chunk receives a high structural prior if and only if it serves as a core constituent (high S^h_{ik}) of a community that strongly aligns with the query.

Weighted Fusion Scoring. Instead of discarding chunks based on the structural prior, we compute a unified relevance score $\mathcal{S}(q, v_i)$ for all chunks simultaneously. This final score fuses the global structural prior with a fine-grained matching function $\mathcal{F}_{hybrid}$:

$$\mathcal{S}(q, v_i) = \gamma \cdot \mathcal{S}_{prior}(q, v_i) + (1 - \gamma) \cdot \mathcal{F}_{hybrid}(q, v_i), \tag{11}$$

where γ is a hyperparameter balancing global topological relevance and local precision. Following standard GraphRAG setting (Guo et al., 2025; Gutiérrez et al., 2025), $\mathcal{F}_{hybrid}(q, v_i)$ integrates the latent dense semantic similarity $\cos(\mathbf{x}_q, \mathbf{x}_i)$ with an explicit lexical-level token score. The fused score combines community-informed relevance with local semantic and lexical matching. Under this formulation, all chunks are scored before the final top-k selection, and the reader receives the selected original passages. The common representation connecting index construction and retrieval is the cumulative affiliation: it is used both to form community prototypes and to transfer query–community relevance to chunk-level scores.

5 Experiments

We evaluate our proposed framework DhiRAG on multiple complex question-answering benchmarks. Our study aims to address the following research questions: RQ1: How does DhiRAG compare to state-of-the-art flat and hierarchical GraphRAG baselines across varying levels of reasoning complexity and contextual noise? RQ2: What are the individual contributions of the topological-semantic synergy (SENet) and the soft assignment paradigm in resolving the HOC dilemma? RQ3: Compared with the RAG baselines, what is the retrieval performance of DhiRAG?

5.1 Experimental Settings

Benchmarks. We evaluate DhiRAG on four benchmarks covering two task categories: single-hop factual question answering and multi-hop question answering. Together, these benchmarks assess the ability to retrieve individual facts and combine evidence distributed across multiple passages. (i) Simple QA. We use PopQA (Mallen et al., 2023) to evaluate factual question answering. Following the evaluation protocol of Jimenez Gutierrez et al. (2024), we randomly sample 1,000 queries from its validation set. This setting assesses whether the retrieval pipeline provides relevant evidence for answering individual factual questions. (ii) Multi-hop QA. We evaluate on HotpotQA (Yang et al., 2018), 2WikiMultiHopQA (Ho et al., 2020), and MuSiQue (Trivedi et al., 2022). These benchmarks assess the ability to combine supporting evidence across multiple passages to derive an answer, complementing the single-hop evaluation on PopQA. For HotpotQA and MuSiQue, we sample 1,000 queries from each benchmark.

Baselines. We compare DhiRAG against a diverse set of baselines, which fall into four categories: (i) zero-shot LLM inference with the GPT-4 series (Achiam et al., 2023); (ii) vanilla retrieval-augmented generation; (iii) knowledge graph–based RAG methods, including HippoRAG (Jimenez Gutierrez et al., 2024), LightRAG (Guo et al., 2025), HippoRAG2 (Gutiérrez et al., 2025), and LinearRAG (Zhuang et al., 2026); (iv) hierarchical graph–based RAG methods, such as RAPTOR (Sarthi et al., 2024), E^{2}GraphRAG (Zhao et al., 2025), ArchRAG (Wang et al., 2026a), and SeRAG (Wei et al., 2026). Specifically, DhiRAG focuses on index construction, so the setup of DhiRAG is a single-step process.

Metrics. Following established evaluation protocols for complex QA (Asai et al., 2024; Wang et al., 2026a; Chen et al., 2026), we report two complementary metrics: (1) *String-based Accuracy*, which

Table 1: Comparison of different RAG methods across three benchmark datasets. The metrics reported are String-based Accuracy (Str-Acc.) and LLM-based Accuracy (LLM-Acc.). Best results are highlighted in bold, and second-best results are underlined.

Model	PopQA		HotpotQA		2WikiMultiHopQA		MuSiQue		Average	
	Str-Acc	LLM-Acc	Str-Acc	LLM-Acc	Str-Acc	LLM-Acc	Str-Acc	LLM-Acc	Str-Acc	LLM-Acc
Direct Zero-shot LLM Inference										
GPT-4o-mini	24.3	23.5	38.2	39.9	37.2	35.9	13.3	14.9	28.3	28.6
GPT-4.1-mini	48.1	44.0	43.6	43.4	54.9	54.4	20.2	22.9	41.7	41.2
Vanilla Retrieval-Augmented Generation										
Vanilla RAG	47.8	43.6	54.2	54.6	47.2	45.0	25.2	28.8	43.6	43.0
Graph-based Retrieval-Augmented Generation Methods										
RAPTOR	32.0	29.1	48.1	46.8	38.2	34.1	21.8	23.9	35.0	33.5
ArchRAG	43.0	27.8	40.9	32.4	55.6	35.9	19.2	16.4	39.7	28.1
E^2GraphRAG	16.0	15.5	49.6	47.3	25.5	20.8	19.2	20.9	27.6	26.1
LightRAG	49.1	46.0	58.9	57.6	63.0	56.3	28.9	26.1	50.0	46.5
HippoRAG	60.7	55.6	60.6	59.0	67.8	60.6	26.0	29.0	53.8	51.1
HippoRAG2	62.1	54.0	61.2	63.1	61.6	59.4	27.7	30.0	53.2	51.6
LinearRAG	51.9	50.5	**64.8**	**65.9**	<u>68.7</u>	<u>62.6</u>	27.9	31.4	53.3	52.6
SeRAG	<u>64.0</u>	<u>57.1</u>	63.5	64.3	65.1	61.5	<u>30.0</u>	<u>31.8</u>	55.7	53.7
DhiRAG	**67.0**	**60.0**	<u>64.2</u>	<u>65.0</u>	**69.1**	**65.3**	**31.9**	**35.2**	**58.1**	**56.4**

measures whether the gold answer is included in the generated answer instead of strictly requiring exact matching; and (2) *LLM-based Accuracy*, an evaluation where an LLM acts as an impartial judge to determine the factual equivalence between the generated answer and the gold reference. Implementation details, including the unified backbone, embedding model, chunk segmentation, and retrieval budget, are provided in Appendix A.1.

5.2 Main Results

Table 1 compares DhiRAG with zero-shot LLM inference, vanilla RAG, and graph-based retrieval methods on four benchmarks: PopQA, HotpotQA, 2WikiMultiHopQA, and MuSiQue. We report both String-based Accuracy (Str-Acc) and LLM-based Accuracy (LLM-Acc), together with their macro-averages across the four datasets.

Overall performance. DhiRAG achieves the highest average performance, with 58.1 Str-Acc and 56.4 LLM-Acc, exceeding SeRAG, the strongest baseline by average accuracy, by 2.4 and 2.7 percentage points, respectively. It ranks first on six of the eight dataset-level metrics and second on the remaining two. Compared with vanilla RAG, DhiRAG improves average Str-Acc and LLM-Acc by approximately 14.5 and 13.4 points. These results support the effectiveness of the complete indexing and retrieval pipeline across the evaluated tasks, while the dataset-specific comparisons reveal where its advantages are most apparent.

Multi-hop question answering. DhiRAG obtains the best results on both 2WikiMultiHopQA and MuSiQue. On 2WikiMultiHopQA, it achieves 69.1 Str-Acc and 65.3 LLM-Acc, improving over LinearRAG, the strongest baseline on this dataset, by 0.4 and 2.7 points. On MuSiQue, DhiRAG reaches 31.9 Str-Acc and 35.2 LLM-Acc, exceeding SeRAG by 1.9 and 3.4 points, respectively. The latter is the largest LLM-Acc improvement over the strongest dataset-specific baseline among the four benchmarks.

The advantage is not uniform across all multi-hop settings. On HotpotQA, DhiRAG obtains 64.2 Str-Acc and 65.0 LLM-Acc, ranking second to LinearRAG by margins of 0.6 and 0.9 points. Thus, the results indicate competitive performance across multi-hop benchmarks rather than universal superiority. The gains on 2WikiMultiHopQA and MuSiQue are consistent with the motivation for combining community-level context with local evidence matching; their specific connection to soft affiliations is examined through the ablation and retrieval-level analyses rather than inferred from answer accuracy alone.

Single-hop factual QA. On PopQA, DhiRAG achieves the highest Str-Acc and LLM-Acc, reaching 67.0 and 60.0, respectively. Relative to SeRAG, the strongest baseline on this dataset, these results represent improvements of 3.0 and 2.9 points. The gains therefore are not restricted to multi-hop

questions: the proposed pipeline also improves end-to-end accuracy on factual queries. Together with the multi-hop results, this shows that incorporating soft hierarchical information can benefit different question types without requiring a separate retrieval architecture for each task.

5.3 Ablation Study

To examine the contribution of each design choice in DhiRAG, we conduct ablation studies on PopQA, HotpotQA, and MuSiQue. We focus on two key aspects of our framework: (i) the topological-semantic synergy introduced by SENet, and (ii) the soft assignment mechanism used for hierarchical indexing. Table 2 reports the results. Removing any of these components consistently degrades performance across all three datasets, indicating that each contributes to the effectiveness of the full model.

Topological-Semantic Synergy. We first disentangle the two sources of information integrated by SENet: graph structure and node semantics. Removing message passing (*w/o Message Passing*) prevents information propagation between neighboring chunks and reduces accuracy by 1.3, 2.5, and 2.1 points on PopQA, HotpotQA, and MuSiQue, respectively. The particularly pronounced drop on HotpotQA is consistent with its multi-hop setting, where answering a question often requires connecting evidence distributed across multiple passages. This result suggests that explicitly propagating information over the chunk graph provides complementary structural context beyond independent semantic representations.

Table 2: Ablation study of DhiRAG on three datasets. We report the *LLM-based Accuracy* (%) for the full model and each ablated variant.

Method	PopQA	HotpotQA	MuSiQue
DhiRAG	60.0	65.0	35.2
Representation Synergy			
w/o Msg. Passing	58.7	62.5	33.1
w/o Node Sem.	57.9	64.2	32.2
Indexing Paradigm			
w/o Soft Assign.	59.3	62.7	33.4

Removing node semantics (*w/o Node Semantics*) leads to drops of 2.1, 0.8, and 3.0 points on PopQA, HotpotQA, and MuSiQue, respectively. Notably, this variant exhibits the largest degradation among all ablations on MuSiQue. In the absence of semantic features, SENet must organize chunks primarily according to graph structure, which provides insufficient information for distinguishing structurally related but semantically different evidence. The larger degradation on MuSiQue suggests that semantic information becomes particularly important when reasoning involves more complex and heterogeneous evidence chains. Together, these two ablations show that topology and semantics play complementary roles: message passing captures structural dependencies between chunks, while node semantics preserves their fine-grained content. Their integration enables SENet to construct representations that remain effective across datasets with different reasoning characteristics.

Effect of Soft Assignments. We next replace the soft assignment mechanism with mutually exclusive cluster assignments (*w/o Soft Assignment*), such that each chunk is associated with only a single cluster at each level of the hierarchy. This modification consistently reduces accuracy by 0.7, 2.3, and 1.8 points on PopQA, HotpotQA, and MuSiQue, respectively. The degradation is especially evident on the two multi-hop benchmarks, indicating that rigid partitioning can discard useful cross-cluster relationships when a chunk is relevant to multiple semantic contexts.

The lower accuracy of the hard-assignment variant indicates that retaining graded affiliations is useful under the evaluated indexing and retrieval protocol. Because changing the assignments also changes community prototypes and structural relevance scores, this ablation measures the contribution of soft affiliation to the complete pipeline. It does not independently identify which semantic facets account for the performance difference.

5.4 Retrieval-Level Evaluation

To decouple retrieval quality from the downstream reader, we directly evaluate the ranked evidence before answer generation. We report two complementary metrics. **Gold Hit@3** measures the percentage of queries for which at least one gold evidence unit appears among the top-three retrieved results, while **First-Gold MRR** evaluates the reciprocal rank of the highest-ranked gold evidence.

The former measures evidence coverage under a restricted retrieval budget, whereas the latter captures ranking quality.

Table 3 yields three main observations. First, DhiRAG achieves the highest Gold Hit@3 on all three datasets: 85.3 on PopQA, 93.4 on HotpotQA, and 82.2 on MuSiQue. Although the margins over the strongest competing method are modest, the consistent improvements under the restrictive top-three budget indicate that the probability-weighted structural prior improves the likelihood that relevant evidence is retained before generation.

Table 3: Retrieval-level evaluation on 1,000 queries per dataset with a retrieval budget of $k = 3$.

Method	PopQA		HotpotQA		MuSiQue	
	Hit@3	MRR	Hit@3	MRR	Hit@3	MRR
LinearRAG	69.3	52.4	92.9	85.9	81.0	68.5
HippoRAG	84.3	73.0	92.4	87.9	79.5	69.4
HippoRAG2	81.4	63.9	92.5	87.2	82.1	68.6
SeRAG	83.4	66.8	89.9	85.0	80.8	70.7
DhiRAG	85.3	68.4	93.4	88.0	82.2	70.9

Second, the benefit is most consistent on the two multi-hop benchmarks. DhiRAG achieves both the highest Gold Hit@3 and the highest First-Gold MRR on HotpotQA and MuSiQue. In particular, compared with SeRAG, the closest token-free hierarchical baseline, DhiRAG improves Gold Hit@3 and First-Gold MRR by 3.5 and 3.0 points on HotpotQA, and by 1.4 and 0.2 points on MuSiQue, respectively. These results suggest that soft hierarchical affiliations are useful not only for including gold evidence in the retrieved set, but also for placing such evidence earlier in the ranking when reasoning requires information across multiple semantic regions.

Third, PopQA exhibits a different pattern. DhiRAG obtains the highest Gold Hit@3, but HippoRAG achieves the highest First-Gold MRR (73.0 versus 68.4). This result suggests that DhiRAG provides broad evidence coverage for factual queries, whereas associative graph retrieval can rank the first relevant fact more aggressively in this single-hop setting. We therefore do not claim uniform superiority in ranking quality across all tasks.

Overall, the retrieval-level results complement the end-to-end results in Table 1. They show that DhiRAG's improvements on the more compositional benchmarks are already observable before generation, reducing the possibility that the reported gains arise solely from reader behavior or LLM-based evaluation. Nevertheless, retrieval metrics and final answer accuracy are not perfectly coupled, as the downstream reader may differ in how effectively it uses the retrieved context.

6 Conclusion

We presented DhiRAG, a framework that connects learned hierarchical indexing and passage-level retrieval through shared soft affiliations. By combining graph message passing with a differentiable structural-entropy objective, DhiRAG learns graded relationships between chunks and communities. The same affiliations support both the construction of community prototypes from chunk embeddings and the transfer of query–community relevance to individual passages. Combining this structural signal with local semantic and lexical matching allows hierarchical context to inform evidence ranking without generating textual community summaries, while retaining original passages as the reader's input. Experiments on PopQA, HotpotQA, 2WikiMultiHopQA, and MuSiQue show that DhiRAG achieves the highest average Str-Acc and LLM-Acc among the evaluated methods, with gains that vary across datasets. Direct retrieval measurements and component ablations provide complementary evidence for the usefulness of the learned index and its non-exclusive affiliations. Together, these findings support learned hierarchical context as a useful complement to local passage matching, rather than a replacement for the original textual evidence.

The framework remains dependent on the quality of its semantic graph and incurs corpus-level optimization costs. Its current static-corpus setting also limits incremental updates. Future work will investigate efficient adaptation to changing corpora and more scalable implementations of affiliation-based retrieval, with particular attention to the trade-off between indexing cost and evidence-ranking quality.

AI USE STATEMENT

We used generative AI tools to assist with manuscript writing and language polishing, including refining phrasing, improving clarity and concision, and organizing the presentation. The authors reviewed and revised all AI-assisted text to ensure consistency with the methodology, experimental results, and intended conclusions. We take full responsibility for the final content of this work, including any text, claims, or artifacts produced with the assistance of generative AI.

REFERENCES

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*, 2023.

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. Self-RAG: Learning to retrieve, generate, and critique through self-reflection. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=hSyW5go0v8.

Shengyuan Chen, Chuang Zhou, Zheng Yuan, Qinggang Zhang, Zeyang Cui, Hao Chen, Yilin Xiao, Jiannong Cao, and Xiao Huang. You don't need pre-built graphs for RAG: Retrieval augmented generation with adaptive reasoning structures. In *The Fortieth AAAI Conference on Artificial Intelligence*, 2026.

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

Zirui Guo, Lianghao Xia, Yanhua Yu, Tu Ao, and Chao Huang. LightRAG: Simple and fast retrieval-augmented generation. In Christos Christodoulopoulos, Tanmoy Chakraborty, Carolyn Rose, and Violet Peng (eds.), *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 10746–10761, Suzhou, China, November 2025. Association for Computational Linguistics. ISBN 979-8-89176-335-7. doi: 10.18653/v1/2025.findings-emnlp.568. URL https://aclanthology.org/2025.findings-emnlp.568/.

Bernal Jiménez Gutiérrez, Yiheng Shu, Weijian Qi, Sizhe Zhou, and Yu Su. From RAG to memory: Non-parametric continual learning for large language models. In *Forty-second International Conference on Machine Learning*, 2025. URL https://openreview.net/forum?id=LWH8yn4HS2.

Taher H Haveliwala. Topic-sensitive pagerank. In *Proceedings of the 11th international conference on World Wide Web*, pp. 517–526, 2002.

Xiaoxin He, Yijun Tian, Yifei Sun, Nitesh Chawla, Thomas Laurent, Yann LeCun, Xavier Bresson, and Bryan Hooi. G-retriever: Retrieval-augmented generation for textual graph understanding and question answering. *Advances in Neural Information Processing Systems*, 37:132876–132907, 2024.

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

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

Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *International Conference on Learning Representations*, 2017. URL https://openreview.net/forum?id=SJU4ayYgl.

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

Angsheng Li. *Science of Artificial Intelligence: Mathematical Principles of Intelligence (In Chinese)*. Science Press, Beijing, 2024.

Angsheng Li and Yicheng Pan. Structural information and dynamical complexity of networks. *IEEE Transactions on Information Theory*, 62(6):3290–3339, 2016.

Rui Li, Zeyu Zhang, Xiaohe Bo, Zihang Tian, Xu Chen, Quanyu Dai, Zhenhua Dong, and Ruiming Tang. CAM: A constructivist view of agentic memory for LLM-based reading comprehension. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2025. URL https://openreview.net/forum?id=ACSOnSHiWe.

Linhao Luo, Zicheng Zhao, Gholamreza Haffari, Dinh Phung, Chen Gong, and Shirui Pan. GFM-RAG: Graph foundation model for retrieval augmented generation. In *The Thirty-ninth Annual Conference on Neural Information Processing Systems*, 2026. URL https://openreview.net/forum?id=0QNmAvQQqj.

Alex Mallen, Akari Asai, Victor Zhong, Rajarshi Das, Daniel Khashabi, and Hannaneh Hajishirzi. When not to trust language models: Investigating effectiveness of parametric and non-parametric memories. In *Proceedings of the 61st annual meeting of the association for computational linguistics (volume 1: Long papers)*, pp. 9802–9822, 2023.

Nils Reimers and Iryna Gurevych. Sentence-bert: Sentence embeddings using siamese bert-networks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 11 2019. URL https://arxiv.org/abs/1908.10084.

Parth Sarthi, Salman Abdullah, Aditi Tuli, Shubh Khanna, Anna Goldie, and Christopher D Manning. RAPTOR: Recursive abstractive processing for tree-organized retrieval. In *The Twelfth International Conference on Learning Representations*, 2024. URL https://openreview.net/forum?id=GN921JHCRw.

Li Sun, Zhenhao Huang, Hao Peng, Yujie Wang, Chunyang Liu, and Philip S. Yu. LSEnet: Lorentz structural entropy neural network for deep graph clustering. In *Forty-first International Conference on Machine Learning*, 2024. URL https://openreview.net/forum?id=L6SRXG92s6.

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

Shu Wang, Yixiang Fang, Yingli Zhou, Xilin Liu, and Yuchi Ma. Archrag: Attributed community-based hierarchical retrieval-augmented generation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 15868–15876, 2026a.

Shurong Wang, Yuqi Pan, Zhuoyang Shen, Meng Zhang, Hongwei Wang, and Guoqi Li. Adaptive hopfield network: Rethinking similarities in associative memory. In *International Conference on Learning Representations*, volume 2026, pp. 116942–116974, 2026b.

Yifan Wei, Yuan Dan, Xiaoyan Yu, and Angsheng Li. Token-free hierarchical indexing for RAG beyond LLM-based summarization. In *Forty-third International Conference on Machine Learning*, 2026. URL https://openreview.net/forum?id=UnNUWCbDO6.

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

Xiaoyan Yu, Yifan Wei, Shuaishuai Zhou, Zhiwei Yang, Li Sun, Hao Peng, Liehuang Zhu, and Philip S Yu. Towards effective, efficient and unsupervised social event detection in the hyperbolic space. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 39, pp. 13106–13114, 2025.

Yaoze Zhang, Rong Wu, Pinlong Cai, Xiaoman Wang, Guohang Yan, Song Mao, Ding Wang, and Botian Shi. Leanrag: Knowledge-graph-based generation with semantic aggregation and hierarchical retrieval. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 40, pp. 34862–34869, 2026.

Yibo Zhao, Jiapeng Zhu, Ye Guo, Kangkang He, and Xiang Li. Eˆ 2graphrag: Streamlining graph-based rag for high efficiency and effectiveness. *arXiv preprint arXiv:2505.24226*, 2025.

Luyao Zhuang, Shengyuan Chen, Yilin Xiao, Huachi Zhou, Yujing Zhang, Hao Chen, Qinggang Zhang, and Xiao Huang. LinearRAG: Linear graph retrieval augmented generation on large-scale corpora. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://openreview.net/forum?id=mCtfkypdm6.

APPENDIX

A TECHNICAL APPENDICES

A.1 IMPLEMENTATION DETAILS

To guarantee a rigorous, reproducible, and equitable evaluation environment, we carefully standardize the foundational infrastructure across DhiRAG and all baseline frameworks.

Unified Backbone Models. We employ `gpt-4o-mini` (Achiam et al., 2023) as the universal Large Language Model backbone, utilizing it consistently for both the downstream reading comprehension (answer generation) and the automated LLM-as-a-judge evaluation. Furthermore, to explicitly isolate the architectural contributions of the retrieval mechanisms and eliminate potential biases stemming from disparate vector representations, we adopt *all-MiniLM-L6-v2* (Reimers & Gurevych, 2019) as the universal embedding model. This lightweight yet highly effective dense encoder is applied strictly across all methods for both initial dense retrieval and graph index construction.

Text Segmentation Strategy. For standard document-based benchmarks (e.g., PopQA, HotpotQA, 2WikiMultihopQA, and MuSiQue), the raw textual corpora are segmented using a fixed chunk size of 1,200 tokens with a sliding window overlap of 100 tokens. This configuration is intentionally selected to preserve localized semantic context while maintaining a manageable graph node granularity.

Retrieval Constraints. Finally, the retrieval budget is strictly constrained to $k = 3$ contexts across all evaluated frameworks during the primary experiments. This restrictive setting deliberately limits the capacity of the context window, allowing us to perform a highly penalized and standardized comparison of context utilization efficiency. By forcing the models to operate under a tight retrieval budget, we can definitively assess each system's capability in semantic density optimization and noise filtration without relying on brute-force context expansion.

A.2 PROMPT CONFIGURATIONS FOR GENERATION AND EVALUATION

To ensure the reproducibility and rigor of our experimental pipeline, we detail the specific prompts utilized for both the answer generation and the automated evaluation phases.

Reading Comprehension and Generation Prompt. Figure 1 illustrates the prompt template employed for the final answer generation. Once DhiRAG completes its probability-weighted single-pass retrieval and isolates the top-k most relevant evidence chunks (typically $k = 3$ in our main experiments), this aggregated context is fed into the generator LLM (`gpt-4o-mini`) alongside the user's query. To maximize the model's multi-hop reasoning capabilities, the prompt enforces a structured Chain-of-Thought (CoT) generation process. It explicitly instructs the model to first articulate its intermediate reasoning steps (prefaced by "Thought:"), methodically connecting the dispersed logical anchors retrieved from overlapping communities. Finally, it requires the model to output a concise, entity-centric response (prefaced by "Answer:") to facilitate standardized evaluation.

Automated Evaluation Prompt (LLM-Acc). Figure 2 presents the prompt utilized to compute the LLM Accuracy (LLM-Acc) metric. In complex multi-hop question-answering benchmarks, relying solely on exact string matching (Str-Acc) can be excessively rigid, often penalizing generated answers that are semantically correct but lexically distinct from the ground truth. To provide a more robust, human-aligned assessment of generation quality, we adopt an LLM-as-a-judge evaluation paradigm. The evaluation prompt instructs a powerful evaluator LLM to meticulously compare the generated prediction against the gold-standard answer based on the original query. The evaluator assesses factual equivalence and logical correctness, outputting a binary judgment that rigorously quantifies DhiRAG's true problem-solving proficiency.

A.3 DATASET STATISTICS

To comprehensively evaluate the retrieval and reasoning capabilities of DhiRAG, we select four diverse benchmarks encompassing single-hop factual knowledge, complex multi-hop reasoning, and

QA Prompt

As an advanced reading comprehension assistant, your task is to analyze text passages and corresponding questions meticulously.

Your response start after "Thought: ", where you will methodically break down the reasoning process, illustrating how you arrive at conclusions.

Conclude with "Answer: " using exact entity names from the source text. Provide a concise, definitive response without further elaboration.

Figure 1: Prompt template utilized for the final QA reading comprehension and answer generation. The prompt enforces a structured reasoning process based on the retrieved top-k chunks.

Evaluation Prompt for LLM Accuracy

You are an expert evaluator.
Please evaluate if the generated answer is correct by comparing it with the gold answer.
Generated answer: {pre_answer}
Gold answer: {gold_ans}

The generated answer should be considered correct if it:
1. Contains the key information from the gold answer
2. Is factually accurate and consistent with the gold answer
3. Does not contain any contradicting information

Respond with ONLY 'correct' or 'incorrect'.
Response:

Figure 2: Prompt template utilized for the automated LLM-as-a-judge evaluation, which computes the LLM Accuracy (LLM-Acc) by assessing the semantic equivalence between the prediction and the ground truth.

long-term conversational contexts. The detailed statistics of our evaluation sets are summarized in Table 4.

Standardized Multi-hop and Factual Benchmarks. To ensure a fair and direct comparison with recent state-of-the-art hierarchical baselines, we strictly follow the evaluation protocols and data splits established by HippoRAG2 (Gutiérrez et al., 2025). Specifically, for PopQA, HotpotQA, 2WikiMultiHopQA, and MuSiQue, we utilize the identical representative subsets, each comprising 1,000 sampled queries paired with their respective candidate passage pools (ranging from 8,676 to 11,656 text passages). This standardized sampling guarantees that performance gains are purely attributable to our Differentiable Hierarchical Indexing mechanism rather than dataset variations.

A.4 Hyperparameter Configurations

To ensure the full reproducibility of our experimental results and facilitate future research on Differentiable Hierarchical Indexing, we meticulously detail the hyperparameter configurations utilized across all phases of the DhiRAG pipeline. Table 5 summarizes the core settings categorized by module: Graph Construction, SENet Optimization, and Retrieval & Generation.

Graph Construction and Segmentation. As discussed in Section A.1, we utilize a chunk size of 1,200 tokens with a sliding window overlap of 100 tokens to balance semantic integrity and com-

Table 4: Statistics of the evaluated benchmark datasets. The evaluation subsets for PopQA, HotpotQA, 2WikiMultiHopQA, and MuSiQue strictly follow the standardized configurations of HippoRAG2 to ensure fair comparison.

Configuration	PopQA	HotpotQA	MuSiQue	2WikiMultiHopQA
Number of Queries	1000	1000	1000	1000
Number of Passages	8676	9221	11656	6119

putational feasibility. The initial topological connections are constructed via a k-Nearest Neighbors (k-NN) graph based on the dense cosine similarity of the *all-MiniLM-L6-v2* embeddings.

SENet Optimization. The Structure Entropy Neural Network (SENet) is optimized end-to-end to learn the soft community assignments. We employ the AdamW optimizer with a learning rate of 1e-3 and a weight decay of 1e-4 to prevent overfitting during the graph convolution phases. The hidden dimensions of the GNN layers are configured to bottleneck the semantic representations, enforcing the extraction of macro-level structural abstractions.

Generation Constraints. For the downstream generative tasks using `gpt-4o-mini`, the temperature is strictly set to 0.0. This deterministic generation setting is crucial to minimize LLM-induced hallucinations and ensure that the performance improvements are entirely attributable to DhiRAG's precise contextual retrieval rather than the stochastic creativity of the generative backbone.

Table 5: Complete hyperparameter configurations for the DhiRAG framework across all standard evaluation benchmarks.

Module	Hyperparameter	Value
Graph Construction	Text Chunk Size	1200
	Chunk Overlap	100
	Embedding Dimension	768
	Initial Graph Sparsity (k-NN neighbors)	10
SENet Optimization	Optimizer	AdamW
	Learning Rate	1e-3
	Weight Decay	1e-4
	GNN Hidden Dimension	768
	Number of GCN Layers	2
	Height of Index Tree	3
	Training Steps	2000
Retrieval & Generation	Dense Retrieval Top-k Budget	3
	LLM Temperature	0.0
	LLM Max Output Tokens	4096
	Weighted Score Weight (γ)	0.2

A.5 Runtime Environment

To ensure complete transparency and reproducibility, all empirical evaluations and training procedures in this study were conducted on a standardized, high-performance hardware infrastructure. The specific machine configuration utilized across all evaluated RAG frameworks is detailed in Table 6.

Table 6: Detailed machine configuration used in our experiments.

Component	Specification
GPU	NVIDIA A100-SXM4-80GB (80GB HBM2e)
CPU	AMD EPYC 7742 64-Core Processor

Table 7: Comparison of indexing time, retrieval time, and token consumption of DhiRAG and baseline methods.

Method	Time (s)		Token Consumption ($\times 10^6$)	
	Indexing	Retrieval (Avg.)	Prompt	Completion
HippoRAG	3322.46	2.112	17.5	2.32
HippoRAG2	1372.39	1.765	10.8	1.50
LinearRAG	942.62	0.408	3.33	0.129
SeRAG	771.06	0.188	3.36	0.116
DhiRAG	939.51	0.433	3.36	0.115

A.6 Time and Computational Overhead

A critical advantage of DhiRAG is its ability to construct hierarchical indices and execute complex reasoning without incurring the prohibitive time and monetary costs associated with traditional LLM-based GraphRAGs. To validate this efficiency, we profile the time overhead of DhiRAG on the HotpotQA dataset (evaluating 1,000 queries over 9,811 passages) utilizing NVIDIA A100 GPUs from our hardware setup. The results are categorized by the pipeline stages and summarized in Table 8.

Graph Construction (Offline Preprocessing). The initial preprocessing phase involves text segmentation, dense embedding extraction via *all-MiniLM-L6-v2*, and Named Entity Recognition (NER) for weighted fusion scoring. On the HotpotQA corpus, this deterministic, purely local preprocessing pipeline completes in approximately 15 minutes.

SENet Optimization (Indexing). This stage highlights the most significant efficiency gain of DhiRAG. Instead of iteratively prompting an LLM to generate discrete community summaries—a process that typically takes hours and incurs massive token costs—DhiRAG optimizes the Structure Entropy Neural Network (SENet) to learn soft assignments and latent abstractions. On HotpotQA, the total end-to-end training time for SENet is merely **78 seconds**. This demonstrates that our continuous structural indexing is exceptionally lightweight and highly scalable.

Retrieval & Generation (Online Inference). During the online inference phase, DhiRAG achieves near real-time responsiveness. The probability-weighted single-pass retrieval mechanism evaluates all structural priors and local features simultaneously, requiring an average of only **0.433 seconds** per query. Subsequently, the downstream generation via `gpt-4o-mini` takes an average of **0.498 seconds**. The combined end-to-end latency per query is strictly sub-second (≈ 0.931 seconds), circumventing the compounding delays typical of multi-hop retrieval and making DhiRAG highly practical for real-time question-answering applications.

Table 8: Empirical time overhead of DhiRAG evaluated on the HotpotQA benchmark. Our continuous indexing via SENet requires only 78 seconds, drastically outperforming the indexing latency of traditional LLM-summarization frameworks.

Pipeline Stage	Metric	Time
Graph Construction	Preprocessing & NER Time	$\approx$ 15 min
SENet Optimization	Total Training Time	78 s
Retrieval & Generation	Average Retrieval Time (per query)	0.433 s
	Average Generation Time (per query)	0.498 s
	Total Inference Latency (per query)	0.931 s

Table 9: Main results with run-to-run variation. Entries report mean ± standard deviation over three runs. Accuracy values are percentages, and standard deviations are expressed in percentage points.

Model	PopQA		HotpotQA		2WikiMultiHopQA		MuSiQue	
	Str-Acc	LLM-Acc	Str-Acc	LLM-Acc	Str-Acc	LLM-Acc	Str-Acc	LLM-Acc
DhiRAG	67.0 ± 0.24	60.0 ± 0.28	64.2 ± 0.24	65.0 ± 0.25	69.1 ± 0.29	65.3 ± 0.17	31.9 ± 0.21	35.2 ± 0.13
SeRAG	64.0 ± 0.21	57.1 ± 0.23	63.5 ± 0.24	64.3 ± 0.27	65.1 ± 0.15	61.5 ± 0.21	30.0 ± 0.29	31.8 ± 0.24
LinearRAG	51.9 ± 0.31	50.5 ± 0.38	64.8 ± 0.22	65.9 ± 0.34	68.7 ± 0.09	62.6 ± 0.05	27.9 ± 0.33	31.4 ± 0.27
HippoRAG2	62.1 ± 0.25	54.0 ± 0.32	61.2 ± 0.16	63.1 ± 0.34	61.6 ± 0.12	59.4 ± 0.19	27.7 ± 0.31	30.0 ± 0.22

B Limitations and Future Work

While DhiRAG successfully resolves the Hierarchical Overlapping Clustering (HOC) dilemma and sets a new standard for multi-hop reasoning, we acknowledge several limitations that provide promising avenues for future research:

Computational Overhead of Graph Optimization. DhiRAG entirely eliminates the prohibitive LLM API costs associated with token-based community summarization. However, this is achieved by shifting the computational paradigm towards training the Structure Entropy Neural Network (SENet). For extremely massive, web-scale corpora, optimizing the continuous structural entropy via graph convolutions may incur higher GPU memory and time overhead during the indexing phase compared to computationally inexpensive, heuristic discrete clustering algorithms (e.g., Louvain or Leiden). Future work will explore scalable graph sampling techniques to accelerate SENet training on billion-scale node graphs.

Handling Dynamic and Streaming Corpora. The current formulation of DhiRAG assumes a static knowledge corpus during the index construction phase. Since the soft community assignments and token-free abstractions are learned via end-to-end global topological optimization, incrementally adding new documents or continually updating existing facts requires either local retraining or computationally intensive graph updates. Developing a dynamic, streaming-compatible differentiable indexing mechanism remains an important next step.

C Additional Analyses

C.1 Run-to-Run Variation

Table 9 reports the mean and standard deviation over three runs for DhiRAG and the principal comparison methods. These statistics describe the observed run-to-run variation under the evaluation protocol. They are not confidence intervals and, by themselves, do not establish statistical significance for differences between methods.

C.2 Boundary-Risk Diagnosis and Overlap Sensitivity

Research question. Does the behavior of hard and soft community affiliations differ in a manner consistent with the proposed HOC mechanism? Answer accuracy alone cannot distinguish the preservation of secondary semantic facets from changes elsewhere in the retrieval and generation pipeline. We therefore conduct two complementary analyses: a post-hoc boundary-risk diagnostic comparing hard and soft affiliations, and a physical-overlap ablation evaluating DhiRAG's downstream behavior.

These analyses address different aspects of the hypothesis. The first evaluates a geometric surrogate associated with community boundaries. The second examines sensitivity to chunk-boundary context. Neither analysis directly measures the recall of independently annotated secondary-topic evidence.

Controlled evaluation protocol. We use PopQA, HotpotQA, and MuSiQue, with 1,000 queries per dataset. Within this study, the chunk length is fixed at $L = 1{,}200$ tokens, and overlap is varied over $o \in \{0, 100, 200\}$ tokens, giving a stride of $L - o$. For each dataset–overlap configuration,

we reconstruct the chunks, embeddings, lexical preprocessing, and hierarchical index, and retrieve three chunks per query for answer generation.

The results are interpreted within this controlled study rather than substituted for the main experimental results. The hard–soft comparison concerns the boundary diagnostic. The downstream overlap ablation evaluates DhiRAG with soft affiliations; it is not a paired hard-versus-soft QA comparison.

C.2.1 Residual-Based Perturbation Model

Inspired by the query-as-variant perspective of Wang et al. (2026b), we describe query variation relative to its associated evidence. For a query–gold-chunk pair with embeddings $(\boldsymbol{q}, \boldsymbol{e})$, define

$$\boldsymbol{r} = \boldsymbol{q} - \boldsymbol{e}. \tag{12}$$

Gold evidence is used only for post-hoc diagnosis and is not provided to SENet training or the retrieval procedure.

Let $\widehat{\boldsymbol{\Sigma}}$ denote the empirical covariance of the centered residuals. For positive total residual variance, we retain the smallest PCA rank p satisfying

$$p = \min\left\{ j : \frac{\sum_{\ell=1}^{j} \lambda_\ell}{\sum_{\ell=1}^{d} \lambda_\ell} \geq 0.95 \right\}, \tag{13}$$

where $\lambda_1 \geq \cdots \geq \lambda_d \geq 0$ are the eigenvalues of $\widehat{\boldsymbol{\Sigma}}$. The truncated covariance is

$$\boldsymbol{\Sigma}_p = \boldsymbol{U}_p \boldsymbol{\Lambda}_p \boldsymbol{U}_p^\top, \qquad \boldsymbol{U}_p \in \mathbb{R}^{d \times p}. \tag{14}$$

Here, p is the PCA rank, not the graph-neighborhood size or the retrieval budget.

The diagnostic uses the covariance-matched perturbation

$$\boldsymbol{\epsilon} = \boldsymbol{U}_p \boldsymbol{\Lambda}_p^{1/2} \boldsymbol{z}, \qquad \boldsymbol{z} \sim \mathcal{N}(\boldsymbol{0}, \boldsymbol{I}_p). \tag{15}$$

If the total residual variance is zero, the perturbation is deterministically zero.

This construction models centered variation around an evidence embedding, not the full distribution of observed queries. In particular, it does not include a possible nonzero mean query–evidence offset, and retaining 95% of total variance does not guarantee the same retained fraction in every boundary-normal direction.

C.2.2 Signed-Margin Boundary Risk

Let $a \in \{\mathrm{hard}, \mathrm{soft}\}$ denote the affiliation scheme. For a fixed reference–competitor comparison associated with $\boldsymbol{e}$, consider the affine diagnostic boundary

$$b_a(\boldsymbol{x}; \boldsymbol{e}) = m_a(\boldsymbol{e}) + \boldsymbol{\Delta}_a(\boldsymbol{e})^\top (\boldsymbol{x} - \boldsymbol{e}). \tag{16}$$

The orientation is chosen such that $b_a < 0$ denotes the competing side. Both the comparison and its normal are held fixed when evaluating the perturbation probability. The signed margin m_a and the normal $\boldsymbol{\Delta}_a$ use the same scale; multiplying both by a positive constant leaves the diagnostic unchanged.

For any specified affine boundary,

$$\boldsymbol{\Delta}_a^\top \boldsymbol{\epsilon} = \left(\boldsymbol{\Lambda}_p^{1/2} \boldsymbol{U}_p^\top \boldsymbol{\Delta}_a \right)^\top \boldsymbol{z}, \tag{17}$$

$$\sigma_a^2(\boldsymbol{e}) = \boldsymbol{\Delta}_a(\boldsymbol{e})^\top \boldsymbol{\Sigma}_p \boldsymbol{\Delta}_a(\boldsymbol{e}). \tag{18}$$

Thus, $\boldsymbol{\Delta}_a^\top \boldsymbol{\epsilon}$ is a zero-mean Gaussian variable with variance $\sigma_a^2(\boldsymbol{e})$.

We retain the abbreviation *HBCP* for the reported diagnostic and define it precisely as the probability of the competing-side event:

$$\mathrm{HBCP}_a(\boldsymbol{e}) := \Pr[b_a(\boldsymbol{e} + \boldsymbol{\epsilon}; \boldsymbol{e}) < 0] \tag{19}$$

$$= \begin{cases} \Phi\left(-\dfrac{m_a(\boldsymbol{e})}{\sigma_a(\boldsymbol{e})} \right), & \sigma_a(\boldsymbol{e}) > 0, \\ \mathbb{I}[m_a(\boldsymbol{e}) < 0], & \sigma_a(\boldsymbol{e}) = 0, \end{cases} \tag{20}$$

Table 10: Boundary-risk diagnostic on HotpotQA and MuSiQue at chunk length $L = 1{,}200$. HBCP values are percentages and refer to the signed-margin probability in Eq. equation 20. Hard and Soft identify the affiliation schemes used for diagnosis, not separate downstream QA experiments. The macro-average weights the six configurations equally.

Dataset	Overlap	Hard HBCP	Soft HBCP
HotpotQA	0	49.42	46.22
	100	48.17	44.08
	200	47.39	43.77
MuSiQue	0	56.15	51.33
	100	57.07	52.06
	200	55.41	50.85
Macro-average	–	52.27	48.05

where Φ is the standard normal cumulative distribution function. The zero-variance case follows from the strict inequality in Eq. equation 19.

Equation equation 20 is exact for the specified affine boundary and Gaussian perturbation model. If the affine boundary approximates a nonlinear decision rule, the resulting probability is a surrogate for that rule rather than its exact error probability.

When $m_a(\boldsymbol{e}) > 0$, the competing-side event represents a crossing from the reference side. When $m_a(\boldsymbol{e}) < 0$, the unperturbed embedding already lies on the competing side, and the probability can exceed 0.5. Accordingly, the statistic is interpreted as a *signed-margin boundary risk*, not unconditionally as a crossing from a correctly assigned starting point. It also concerns one selected comparison, not the probability of crossing any community boundary.

For diagnostic pairs $\mathcal{P}$ with normalized nonnegative aggregation weights, define

$$\overline{\mathrm{HBCP}}_a = \sum_{(q,e)\in\mathcal{P}} w_{q,e}\,\mathrm{HBCP}_a(\boldsymbol{e}), \qquad \sum_{(q,e)\in\mathcal{P}} w_{q,e} = 1. \tag{21}$$

The hard-to-soft difference is reported in percentage points:

$$\Delta_{\mathrm{HBCP}} = 100\left(\overline{\mathrm{HBCP}}_{\mathrm{hard}} - \overline{\mathrm{HBCP}}_{\mathrm{soft}}\right). \tag{22}$$

C.2.3 Diagnostic Results

Comparison of hard and soft affiliations. Table 10 reports lower HBCP for soft affiliations in all six dataset–overlap conditions. The reductions range from 3.20 to 5.01 percentage points. Averaging the six conditions equally gives a decrease from 52.27% to 48.05%, corresponding to a reduction of 4.22 percentage points. These values describe a lower modeled probability of the selected competing-side event under soft affiliations. They should not be interpreted as an equivalent reduction in retrieval-error rate.

The difference is also present at zero window overlap: 3.20 points on HotpotQA and 4.82 points on MuSiQue. Thus, the observed hard–soft difference is not restricted to configurations with duplicated boundary tokens. However, its magnitude does not increase monotonically with overlap on either dataset. The experiment therefore does not establish that physical overlap is a quantitative proxy for semantic topic mixing or that increasing overlap necessarily increases the benefit of soft affiliations.

C.2.4 Downstream Overlap Sensitivity

Effect on downstream accuracy. Table 11 shows higher downstream accuracy with nonzero overlap on both evaluated datasets. On HotpotQA, increasing overlap from 0 to 100 improves Str-Acc from 61.7 to 63.6 and LLM-Acc from 61.9 to 64.9. Increasing overlap further to 200 yields 64.0 Str-Acc and 64.8 LLM-Acc. On MuSiQue, Str-Acc increases from 29.5 to 31.9 and 32.0, while LLM-Acc increases from 31.9 to 35.1 and 35.6, respectively.

The larger observed gains occur between $o = 0$ and $o = 100$; increasing overlap from 100 to 200 produces smaller changes in both metrics. These observations are consistent with boundary

Table 11: Physical-overlap ablation of DhiRAG with soft affiliations on HotpotQA and MuSiQue. All configurations use $L = 1{,}200$ and retrieve three chunks. Accuracy values are percentages. These controlled-study results are reported separately from the main evaluation.

Dataset	Overlap	# Chunks	Str-Acc	LLM-Acc
HotpotQA	0	984	61.7	61.9
	100	1073	63.6	64.9
	200	1180	64.0	64.8
MuSiQue	0	1016	29.5	31.9
	100	1108	31.9	35.1
	200	1219	32.0	35.6

context being useful in the evaluated multi-hop settings. However, changing overlap also changes the indexed chunks, their representations, and the resulting graph. The QA differences therefore cannot be attributed solely to the preservation of a particular secondary semantic facet.

Index-size trade-off. The nominal overlap ratio is o/L: 8.33% at $o = 100$ and 16.67% at $o = 200$. This is distinct from the increase in indexed text relative to non-overlapping windows. For a sufficiently long token sequence, ignoring end effects, the approximate relative overhead is

$$\frac{L}{L-o} - 1 = \frac{o}{L-o}, \tag{23}$$

which equals 9.09% and 20.0%, respectively.

On both datasets, the observed chunk counts increase by approximately 9% at $o = 100$ and 20% at $o = 200$, relative to $o = 0$. These counts quantify the change in index nodes, not the complete runtime or memory overhead. Within this study, $o = 100$ provides an intermediate accuracy–size operating point: it captures most of the observed QA gain while using fewer chunks than $o = 200$. It is not the best setting for every dataset or metric.

Interpretation and scope. The boundary diagnostic and the overlap ablation provide complementary, but distinct, evidence. The former compares a model-dependent geometric risk under hard and soft affiliations; the latter measures sensitivity to physical redundancy within the soft-assignment pipeline. Neither experiment directly estimates secondary-evidence recall, and physical overlap does not independently control topic composition.

Accordingly, the supported conclusion is that soft affiliations exhibit lower reported boundary risk under the specified diagnostic, while moderate physical overlap is associated with higher downstream accuracy in the evaluated HotpotQA and MuSiQue settings. These observations are consistent with the motivation for mitigating boundary-related information loss, but do not establish a general causal guarantee of semantic preservation.