AI-H^{2}RAG: ASYMMETRIC INDEXING-BASED HIERARCHICAL HYPER-RAG

Anonymous authors
Paper under double-blind review

ABSTRACT

Structured RAG has progressed from modeling pairwise relations with ordinary graphs to representing multi-way, higher-order relations with hypergraphs. However, existing approaches still model cross-granularity hierarchies only to a limited extent, constraining their ability to integrate global context with local facts. Inspired by hierarchical associative memory in the cortico-hippocampal system, we note that human memory jointly encodes multiple elements to link higher-level representations with concrete details. Partial cues can then rapidly reinstate relevant context and facts, with high-level context guiding local interpretation and low-level details grounding holistic recall. Following this principle, we propose Asymmetric Indexing-based Hierarchical Hyper-RAG (AI-H^{2}RAG). AI-H^{2}RAG constructs three-layer nested topic–chunk, chunk–sentence, and sentence–entity hypergraphs, organizing evidence so that retrieval can proceed from topic-level scope to entity-level facts and trace local facts back to their topical context. Its asymmetric index propagates final parent context downward and aggregates raw child evidence upward. Each query is then aligned with the index to progressively narrow the candidate space and expand linked evidence for answer generation. Evaluations span eight document corpora and include five generator backbones and three evidence-depth stages. AI-H^{2}RAG achieves the highest average answer quality across the evaluated settings, with relative gains of up to 7.6% over the strongest baseline. Its online retrieval latency is lower than that of the strongest structured-retrieval baseline, showing that hierarchical higher-order retrieval improves answer quality without sacrificing efficiency.

1 INTRODUCTION

Retrieval-augmented generation (RAG) augments generation with external evidence retrieved at inference time (Lewis et al., 2020). By decoupling knowledge from model parameters, it supports source-grounded answers and updates to evolving corpora without retraining. These properties are particularly valuable in specialized domains, where answers depend on document-specific terminology, constraints, and time-varying evidence (Yang et al., 2025). Accordingly, RAG has emerged as a widely adopted paradigm in both academic research and industrial applications.

Current RAG methodologies broadly retrieve evidence from documents, graphs, or hypergraphs. Document-based RAG independently indexes text chunks and retrieves them through learned query–passage similarity (Izacard et al., 2023).Graph-based RAG organizes entities and relations into structured retrievers, connecting distributed evidence through entity paths or aggregating support within communities (Jin et al., 2024; Zhong et al., 2023). Hypergraph-based RAG further represents higher-order relations among multiple entities or facts (Feng et al., 2019; 2026). However, none of these paradigms explicitly captures the hierarchical document structure.

Human memory offers a useful computational analogy for hierarchical knowledge organization. The cortico-hippocampal system associates multiple elements, links higher-level representations with concrete details, and reinstates relevant context from partial cues. Documents exhibit a comparable organization across topics, chunks, sentences, and entities and answering complex questions requires combining global context with local facts. Hypergraphs naturally represent this hierarchy because upper-level semantic units define hyperedges over lower-level units while permitting overlapping memberships across levels (Chien et al., 2022).

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.

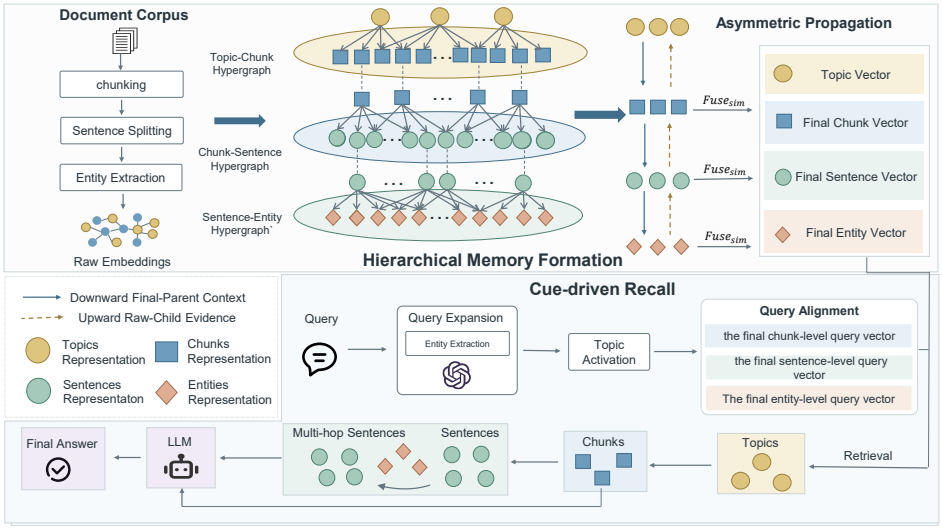


Figure 1: Overview of AI-H^{2}RAG. The framework comprises two coupled processes: hierarchical memory formation and cue-driven recall.

Guided by this principle, we develop AI-H^{2}RAG, a bio-inspired framework that casts document retrieval as two coupled processes: hierarchical memory formation and cue-driven recall. It links higher-level context with lower-level factual details through three-layer nested hypergraphs, while propagating contextual guidance downward and raw factual evidence upward. Queries follow the same hierarchy to progressively narrow the search space and expand linked evidence for answer generation.

We evaluate AI-H^{2}RAG on eight knowledge-intensive document corpora spanning five machine-tool and three general domains (Guo et al., 2025). Additional evaluations cover five generator backbones and three evidence-depth stages on MechanicalCorp, testing robustness across models, domains, and question complexity. AI-H^{2}RAG achieves the highest average answer quality across the evaluated settings and maintains lower online retrieval latency compared to the strongest structured-retrieval baseline. Together, these results support its practical value for accurate and efficient question answering across diverse real-world document settings.

Our contributions are summarized as follows:

- **Bio-inspired hierarchical memory formation.** We construct three-layer nested hypergraphs that bind topics, chunks, sentences, and entities into multi-way relational units. This organization connects higher-level contextual representations with local factual details while preserving their cross-level membership relations.
- **Asymmetric hypergraph retrieval network.** We build an asymmetric retrieval network over the nested hypergraphs to construct the document index. Raw child evidence is aggregated upward to support higher-level semantics, while final parent context is propagated downward to guide finer-grained representations. This network connects hierarchical memory formation with evidence retrieval.
- **Cue-activated hierarchical evidence recall.** We formulate query answering as a recall process over the hypergraph retrieval network, with each query serving as a partial cue. Topic activation progressively narrows the search from abstract themes to concrete evidence, while shared entities trace related sentences. This process supports fast evidence recovery without scanning all document units globally.

2 Related Work

RAG Retrieval-augmented generation (RAG) couples a retriever with a language model to provide external evidence for knowledge-intensive tasks (Guu et al., 2020). Scaled retrieval and self-

reflective variants extend this paradigm (Asai et al., 2024; Borgeaud et al., 2022). Recent surveys summarize the rapidly expanding RAG literature (Gao et al., 2024; Huang & Huang, 2026). Dense retrieval improves evidence access through learned query and passage representations. Query rewriting and hypothetical-document expansion improve underspecified retrieval inputs (Ma et al., 2023). Question-answering benchmarks such as SQuAD and Natural Questions evaluate evidence-grounded answering (Rajpurkar et al., 2016; Kwiatkowski et al., 2019). RAG evaluation also assesses retrieval grounding and generated responses (Es et al., 2024). However, conventional RAG usually indexes passages independently, limiting its ability to coordinate document scope with local factual evidence (Adjali et al., 2024; Liu et al., 2024).

Graph-based RAG Graph-based RAG connects related evidence through entities, triples, and communities (Pan et al., 2024; Peng et al., 2025). GraphRAG supports local and global query-focused summarization through entity graphs and community reports (Edge et al., 2025). LightRAG improves retrieval efficiency with dual-level graph retrieval (Guo et al., 2025). Knowledge-graph methods also support relation-path reasoning for complex question answering (Lan et al., 2023; Xu et al., 2024). Graph neural encoders aggregate neighborhood information through convolution or sampled message passing (Kipf & Welling, 2017; Hamilton et al., 2017). Attention-based graph encoders provide an alternative mechanism for weighting neighborhood information (Veličković et al., 2018; Wu et al., 2021). Entity linking resolves surface mentions to canonical graph nodes (Hoffart et al., 2011; Sevgili et al., 2022), while entity-aware contextual encoders provide learned entity representations (Yamada et al., 2020). Nevertheless, these methods primarily capture pairwise relations, entity neighborhoods, or coarse summaries, failing to explicitly construct the hierarchical document structure required for multi-granularity retrieval in QA.

Hypergraph-based RAG Hypergraphs generalize ordinary graphs by allowing each hyperedge to connect multiple vertices, thereby representing higher-order dependencies beyond pairwise links (Feng et al., 2019; Gao et al., 2023). Set-based hypergraph architectures provide a complementary formulation for aggregating variable-sized neighborhoods (Chien et al., 2022). Hyper-RAG extends this principle to retrieval-augmented generation through multi-entity hypergraph organization (Feng et al., 2026). Existing hypergraph RAG methods, however, typically construct hyperedges over entities or other single-level units. They do not employ hypergraphs to explicitly represent the hierarchical structure of documents, which is explored in this work.

3 METHOD

AI-H^{2}RAG follows a bio-inspired memory cycle that couples hierarchical memory formation with cue-activated recall, as shown in Figure 1. It constructs an asymmetric retrieval network on the constructed nested hypergraph memory and aligns each query with this network for coarse-to-fine evidence retrieval and answer generation.

3.1 BIO-INSPIRED HIERARCHICAL MEMORY FORMATION WITH NESTED HYPERGRAPHS

Documents organize evidence across topics, chunks, sentences, and entities. To represent document levels, we propose a hierarchical hypergraph memory method that includes multi-level knowledge extraction and hypergraph binding, cross-level membership encoding, and multi-level representation initialization. The overall architecture is illustrated in Figure 2.

Multi-level Knowledge Extraction and Hypergraph Binding. Given a document corpus $\mathcal{D} = \{d_n\}_{n=1}^{N_\mathcal{D}}$, we decompose each document into chunks, sentences, and entities. Across the corpus, these units form the collections $\mathcal{C} = \{c_i\}_{i=1}^{N}$, $\mathcal{S} = \{s_j\}_{j=1}^{J}$, and $\mathcal{U} = \{u_m\}_{m=1}^{M}$, respectively. Here, we use sentences to refer to sentence-level textual units, and employ the term entities broadly to encompass both conventional entities and salient semantic elements, including actions, conditions, events, and relations. During entity extraction, we additionally assign each extracted entity an importance score. The prompts for extracting sentences and entities are provided in

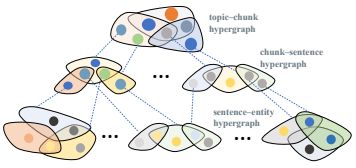


Figure 2: Hierarchical hypergraph memory construction.

Appendix A and Appendix B. To represent these textual units and their relationships, a hypergraph is defined as $\mathcal{H} = (\mathcal{V}, \mathcal{E})$, where $\mathcal{V}$ is the vertex set and $\mathcal{E}$ is the hyperedge set.

A fuzzy topic–chunk hypergraph is first constructed. To obtain topic representations, we induce a set of latent topic anchors $\mathcal{T} = \{t_k\}_{k=1}^{K}$ and a fuzzy topic–chunk membership matrix $\mathbf{M}^{\mathrm{T}\to\mathrm{C}} = [m_{ki}^{\mathrm{T}\to\mathrm{C}}] \in [0,1]^{K\times N}$ from the chunk set $\mathcal{C}$, where $m_{ki}^{\mathrm{T}\to\mathrm{C}} \in [0,1]$ denotes the membership strength of chunk c_i with respect to topic t_k. Unlike hard topic partitioning, this fuzzy assignment preserves cross-topic semantic associations by allowing a chunk to participate in multiple topic hyperedges. The detailed fuzzy-topic induction procedure is provided in Appendix C.

Each topic t_k serves as the semantic anchor of a topic-induced hyperedge $e_k^{\mathrm{T}\to\mathrm{C}} = \{c_i \in \mathcal{C} \mid m_{ki}^{\mathrm{T}\to\mathrm{C}} > 0\}$. Accordingly, with $\mathcal{E}^{\mathrm{T}\to\mathrm{C}} = \{e_k^{\mathrm{T}\to\mathrm{C}}\}_{k=1}^{K}$, we define the topic–chunk hypergraph as $\mathcal{H}^{\mathrm{T}\to\mathrm{C}} = \left(\mathcal{C}, \mathcal{E}^{\mathrm{T}\to\mathrm{C}}\right)$. We further construct two containment-based hypergraph layers: $\mathcal{H}^{\mathrm{C}\to\mathrm{S}} = \left(\mathcal{S}, \mathcal{E}^{\mathrm{C}\to\mathrm{S}}\right), \mathcal{H}^{\mathrm{S}\to\mathrm{U}} = \left(\mathcal{U}, \mathcal{E}^{\mathrm{S}\to\mathrm{U}}\right)$, where each chunk c_i is associated with a hyperedge $e_i^{\mathrm{C}\to\mathrm{S}} \in \mathcal{E}^{\mathrm{C}\to\mathrm{S}}$ that connects its contained sentences, and each sentence s_j is associated with a hyperedge $e_j^{\mathrm{S}\to\mathrm{U}} \in \mathcal{E}^{\mathrm{S}\to\mathrm{U}}$ that connects its extracted entities. Consequently, the three-layer nested hypergraph layers represent four document levels: topics, chunks, sentences, and entities.

Cross-level Membership Encoding. We preserve membership relations between adjacent document levels through three incidence mappings: $\mathbf{M}^{\mathrm{T}\to\mathrm{C}} = [m_{ki}^{\mathrm{T}\to\mathrm{C}}] \in [0,1]^{K\times N}$, $\mathbf{M}^{\mathrm{C}\to\mathrm{S}} = [m_{ij}^{\mathrm{C}\to\mathrm{S}}] \in \{0,1\}^{N\times J}$, and $\mathbf{M}^{\mathrm{S}\to\mathrm{U}} = [m_{jm}^{\mathrm{S}\to\mathrm{U}}] \in [0,1]^{J\times M}$, where N, K, J, and M denote the numbers of chunks, topics, sentences, and entities, respectively. The topic–chunk mapping retains fuzzy membership weights, whereas the remaining mappings encode document containment relations:

$$m_{ij}^{\mathrm{C}\to\mathrm{S}} = \mathbb{I}[s_j \subset c_i], \qquad m_{jm}^{\mathrm{S}\to\mathrm{U}} = \rho_m\, \mathbb{I}[u_m \subset s_j], \tag{1}$$

where $\rho_m \in [0,1]$ is the importance score assigned to entity u_m during entity extraction. These mappings allow retrieval to move structurally in both directions. A fine-grained entity can be traced back to its enclosing sentence, chunk, and topic, while a coarse topic can look for local facts that support it.

Multi-level Representation Initialization. Before cross-level information propagation, we initialize an ℓ_2-normalized raw representation for every document unit. Let $\mathrm{Enc}_{\mathrm{bge}}(\cdot)$ denote the BGE-M3 text encoder (Chen et al., 2024) and $\mathrm{norm}(\cdot)$ denote ℓ_2 normalization. For chunks, sentences, and entities, we obtain:

$$\mathbf{h}_{c_i}^{(\mathrm{raw})} = \mathrm{norm}(\mathrm{Enc}_{\mathrm{bge}}(c_i)), \quad \mathbf{h}_{s_j}^{(\mathrm{raw})} = \mathrm{norm}(\mathrm{Enc}_{\mathrm{bge}}(s_j)), \quad \mathbf{h}_{u_m}^{(\mathrm{raw})} = \mathrm{norm}(\mathrm{Enc}_{\mathrm{bge}}(u_m)). \tag{2}$$

The initial representation of topic t_k is obtained by aggregating its member chunks according to their fuzzy memberships:

$$\mathbf{h}_{t_k}^{(\mathrm{raw})} = \mathrm{norm}\left(\frac{\sum_{c_i \in e_k^{\mathrm{T}\to\mathrm{C}}} m_{ki}^{\mathrm{T}\to\mathrm{C}} \mathbf{h}_{c_i}^{(\mathrm{raw})}}{\sum_{c_i \in e_k^{\mathrm{T}\to\mathrm{C}}} m_{ki}^{\mathrm{T}\to\mathrm{C}}}\right). \tag{3}$$

These representations preserve the original information available at each level.

3.2 Asymmetric Hypergraph Retrieval Network

After constructing the hierarchical hypergraph, we introduce an asymmetric hypergraph retrieval network that constructs final index representations across document levels through cross-level information propagation during index construction, as illustrated in Figure 3. The mechanism comprises asymmetric context-evidence propagation and layer-wise retrieval network construction.

Asymmetric Context–Evidence Propagation. For notational convenience, we define similarity-weighted fusion between a reference representation $\mathbf{r}$ and an input set $\mathcal{Z} = \{\mathbf{z}_i\}_{i=1}^{L}$ as

$$\alpha_i = \frac{\exp(\mathbf{r}^\top \mathbf{z}_i/\kappa)}{\sum_{j=1}^{L} \exp(\mathbf{r}^\top \mathbf{z}_j/\kappa)}, \qquad \mathrm{Fuse}_{\mathrm{sim}}(\mathbf{r}, \mathcal{Z}) = \mathrm{norm}\left(\sum_{i=1}^{L} \alpha_i \mathbf{z}_i\right), \tag{4}$$

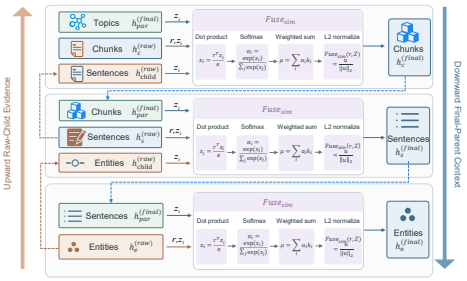


Figure 3: Asymmetric propagation. Final parent context propagates downward, while raw child evidence propagates upward.

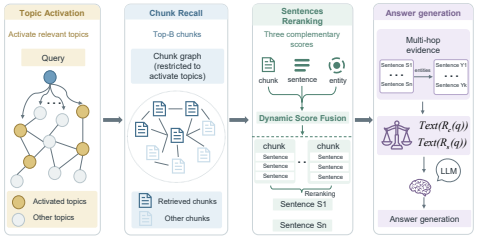


Figure 4: Coarse-to-fine retrieval and generation. Activated topics guide coarse-to-fine evidence retrieval for answer generation.

where κ is a temperature parameter. For an element x, let $\mathrm{Par}(x)$ and $\mathrm{Child}(x)$ denote its associated elements at the immediately coarser and finer levels, respectively. Each element $x \in \mathcal{C} \cup \mathcal{S} \cup \mathcal{U}$ is then updated by

$$\mathbf{h}_x^{(\mathrm{final})} = \mathrm{Fuse}_{\mathrm{sim}}\left(\mathbf{h}_x^{(\mathrm{raw})}, \left\{\mathbf{h}_x^{(\mathrm{raw})}, \mathbf{h}_{\mathrm{par}}^{(\mathrm{final})}(x), \mathbf{h}_{\mathrm{child}}^{(\mathrm{raw})}(x)\right\}\right). \tag{5}$$

where $\mathbf{h}_{\mathrm{par}}^{(\mathrm{final})}(x)$ and $\mathbf{h}_{\mathrm{child}}^{(\mathrm{raw})}(x)$ denote the aggregated final parent context and raw child evidence of x, respectively. When the corresponding associated set is non-empty, they are computed as

$$\mathbf{h}_{\mathrm{par}}^{(\mathrm{final})}(x) = \mathrm{norm}\left(\frac{\sum_{p\in\mathrm{Par}(x)} w_{p,x}\mathbf{h}_p^{(\mathrm{final})}}{\sum_{p\in\mathrm{Par}(x)} w_{p,x}}\right), \quad \mathbf{h}_{\mathrm{child}}^{(\mathrm{raw})}(x) = \mathrm{norm}\left(\frac{\sum_{y\in\mathrm{Child}(x)} w_{x,y}\mathbf{h}_y^{(\mathrm{raw})}}{\sum_{y\in\mathrm{Child}(x)} w_{x,y}}\right). \tag{6}$$

The weights are inherited from the hierarchical membership mappings: $w_{t_k,c_i} = m_{ki}^{\mathrm{T}\to\mathrm{C}}$, $w_{c_i,s_j} = m_{ij}^{\mathrm{C}\to\mathrm{S}}$, and $w_{s_j,u_m} = m_{jm}^{\mathrm{S}\to\mathrm{U}}$. Only elements with nonzero membership or containment weights participate in the corresponding aggregation.

The raw representation $\mathbf{h}_x^{(\mathrm{raw})}$ serves as the fusion reference and is also included explicitly in the fusion inputs, thereby anchoring the resulting representation to the original semantics of x. The propagation is asymmetric: final representations from the coarser level provide downward contextual guidance, whereas only raw representations from the finer level provide upward factual support. Thus, contextual information received by a finer-level element is never reused as factual evidence at a coarser level. For entity elements without finer-level children, $\mathbf{h}_{\mathrm{child}}^{(\mathrm{raw})}(x)$ is omitted from the fusion inputs.

Layer-wise Retrieval Network Construction. Starting from the raw representations initialized in **Multi-level Representation Initialization** of Section 3.1, we use the cross-level incidence mappings $\mathbf{M}^{\mathrm{T}\to\mathrm{C}}$, $\mathbf{M}^{\mathrm{C}\to\mathrm{S}}$, and $\mathbf{M}^{\mathrm{S}\to\mathrm{U}}$ to construct the final hierarchical index representations. No additional propagation is performed at the topic level; the initialized topic representations are directly retained as the final representations: $\mathbf{h}_{t_k}^{(\mathrm{final})} = \mathbf{h}_{t_k}^{(\mathrm{raw})}$.

The chunk, sentence, and entity representations are updated sequentially in a coarse-to-fine order according to Equation 5. Each chunk integrates final topic context with raw sentence evidence, and each sentence integrates final chunk context with importance-weighted raw entity evidence. Each entity integrates its raw representation with the final context of its parent sentence, without a child-evidence term. This procedure yields the final topic, chunk, sentence, and entity representations $\{\mathbf{h}_{t_k}^{(\mathrm{final})}\}_{k=1}^{K}$, $\{\mathbf{h}_{c_i}^{(\mathrm{final})}\}_{i=1}^{N}$, $\{\mathbf{h}_{s_j}^{(\mathrm{final})}\}_{j=1}^{J}$, and $\{\mathbf{h}_{u_m}^{(\mathrm{final})}\}_{m=1}^{M}$, respectively. Appendix D provides the corresponding layer-specific aggregation and update formulas. We further find that incorporating hierarchical context during propagation enables the resulting entity representations to distinguish the same entity when it exhibits different meanings across document contexts. A diagnostic analysis is provided in Appendix E.

3.3 CUE-ACTIVATED HIERARCHICAL EVIDENCE RECALL AND GENERATION

To retrieve evidence at the granularity required for question answering, we develop a evidence recall and generation strategy comprising partial-cue query alignment, coarse-to-fine evidence retrieval and expansion, and evidence-grounded answer generation, as illustrated in Figure 4.

Partial-Cue Query Alignment. Given the hierarchically indexed document structure, we align each query with the corresponding document levels. For a query q, an LLM is used first to construct an expanded textual query $\tilde{q}$ (Ma et al., 2023) and extract its entities $\mathcal{U}_q = \{u_\ell^q\}_{\ell=1}^{L_q}$. Each extracted entity u_ℓ^q is assigned an importance score $\rho_\ell^q \in [0, 1]$. Following the raw-representation initialization in Equation 2, we encode $\tilde{q}$ and its entities as $\mathbf{h}_q^{(\text{raw})}$ and $\{\mathbf{h}_{u_\ell^q}^{(\text{raw})}\}_{\ell=1}^{L_q}$, respectively.

The raw query representation is adopted to activate relevant topic index representations as:

$$a_{qk} = \mathbf{h}_q^{(\text{raw})\top} \mathbf{h}_{t_k}^{(\text{final})}, \qquad \mathcal{A}(q) = \text{TopP}_{t_k \in \mathcal{T}}\, a_{qk}, \tag{7}$$

where $\mathcal{A}(q)$ is the set of activated topics. Then, we construct query-side representations that correspond to the chunk, sentence, and entity levels of the hierarchical index: $\mathbf{h}_{q,\text{C}}^{(\text{final})}$, $\mathbf{h}_{q,\text{S}}^{(\text{final})}$, and $\{\mathbf{h}_{u_\ell^q}^{(\text{final})}\}_{\ell=1}^{L_q}$. To this end, two query signals are designed at the topic level and the entity level as:

$$\mathbf{h}_{\text{topic}}^{(\text{final})}(q) = \text{norm}\left(\frac{\sum_{t_k \in \mathcal{A}(q)} \beta_{qk} \mathbf{h}_{t_k}^{(\text{final})}}{\sum_{t_k \in \mathcal{A}(q)} \beta_{qk}}\right), \quad \mathbf{h}_{\text{entity}}^{(\text{raw})}(q) = \text{norm}\left(\frac{\sum_{\ell=1}^{L_q} \rho_\ell^q \mathbf{h}_{u_\ell^q}^{(\text{raw})}}{\sum_{\ell=1}^{L_q} \rho_\ell^q}\right), \tag{8}$$

where $\beta_{qk} = \exp(a_{qk}/\tau_q)$ is the temperature-scaled weight of the activation score a_{qk}, τ_q is a temperature parameter, and ρ_ℓ^q denotes the importance score of query entity u_ℓ^q. Following the asymmetric fusion principle used for index construction, we uses these signals to construct query-side representations aligned with the chunk, sentence, and entity levels of the document index:

$$\begin{aligned}
\mathbf{h}_{q,\text{C}}^{(\text{final})} &= \text{Fuse}_{\text{sim}}\left(\mathbf{h}_q^{(\text{raw})}, \left\{\mathbf{h}_q^{(\text{raw})}, \mathbf{h}_{\text{topic}}^{(\text{final})}(q)\right\}\right), \\
\mathbf{h}_{q,\text{S}}^{(\text{final})} &= \text{Fuse}_{\text{sim}}\left(\mathbf{h}_q^{(\text{raw})}, \left\{\mathbf{h}_q^{(\text{raw})}, \mathbf{h}_{q,\text{C}}^{(\text{final})}, \mathbf{h}_{\text{entity}}^{(\text{raw})}(q)\right\}\right), \\
\mathbf{h}_{u_\ell^q}^{(\text{final})} &= \text{Fuse}_{\text{sim}}\left(\mathbf{h}_{u_\ell^q}^{(\text{raw})}, \left\{\mathbf{h}_{u_\ell^q}^{(\text{raw})}, \mathbf{h}_{q,\text{S}}^{(\text{final})}\right\}\right).
\end{aligned} \tag{9}$$

Coarse-to-Fine Evidence Retrieval and Expansion. Retrieval proceeds along the same document hierarchy in a coarse-to-fine manner. The activated topics first restrict the candidate chunks through the topic–chunk mapping:

$$\mathcal{C}(q) = \left\{c_i \in \mathcal{C} \;\middle|\; \exists t_k \in \mathcal{A}(q),\ m_{ki}^{\text{T}\to\text{C}} > 0\right\}. \tag{10}$$

Coarse chunk recall is performed by matching each candidate chunk with the chunk-aligned query representation:

$$r_{c_i} = \mathbf{h}_{q,\text{C}}^{(\text{final})\top} \mathbf{h}_{c_i}^{(\text{final})}. \tag{11}$$

Top-B chunks are selected and use $\mathbf{M}^{\text{C}\to\text{S}}$ to define the sentence candidate set. We subsequently rerank candidate sentences using three complementary relevance signals:

$$r_{j,\text{C}} = \mathbf{h}_q^{(\text{raw})\top} \mathbf{h}_{c_i}^{(\text{final})}, \qquad r_{j,\text{S}} = \mathbf{h}_q^{(\text{raw})\top} \mathbf{h}_{s_j}^{(\text{final})}, \qquad r_{j,\text{U}} = \frac{1}{L_q} \sum_{\ell=1}^{L_q} \max_{u_m \in \mathcal{U}(s_j)} \mathbf{h}_{u_\ell^q}^{(\text{final})\top} \mathbf{h}_{u_m}^{(\text{final})}, \tag{12}$$

where c_i is the retrieved chunk associated with candidate sentence s_j, and $\mathcal{U}(s_j) = \{u_m \in \mathcal{U} \mid m_{jm}^{\text{S}\to\text{U}} > 0\}$ denotes the set of entities extracted from s_j. Appendix F justifies our choice of query representations for the chunk-level and sentence-level signals in sentence reranking. The three signals measure chunk-level context relevance, direct sentence relevance, and entity-level relevance, respectively. They are adaptively combined for fine-grained evidence selection:

$$\boldsymbol{\omega}_j = \text{softmax}\left(\frac{[r_{j,\text{C}}, r_{j,\text{S}}, r_{j,\text{U}}]}{\tau_\text{r}}\right), \qquad R_j = \sum_{g \in \{\text{C},\text{S},\text{U}\}} \omega_{j,g} r_{j,g}, \tag{13}$$

where τ_r is a temperature parameter. The highest-ranked sentences provide precise local evidence. Starting from their extracted entities, we further expand to linked sentences with overlapping relevant entities, thereby recovering multi-hop evidence beyond the initially retrieved chunks. Details of multi-hop evidence expansion procedure are provided in Appendix G.

Evidence-Grounded Answer Generation. Let $\mathcal{R}_C(q)$ denote the top-ranked chunk evidence and $\mathcal{R}_S(q)$ denote the reranked and expanded sentence evidence. The answer is generated as

$$\hat{y} = \mathrm{LLM}(q,\ \mathrm{Text}(\mathcal{R}_C(q))\ ,\ \mathrm{Text}(\mathcal{R}_S(q)))\ , \tag{14}$$

where chunk evidence provides document-level context and sentence evidence provides fine-grained factual support.

4 Experiments

4.1 Experimental Setup

We evaluate eight document corpora, including five machine-tool corpora and three general-domain corpora (i.e., MixCorp, PathologyCorp, and PhysicsCorp). Among these corpora, MechanicalCorp supports a three-stage evidence-depth evaluation: Stage 1 requires direct evidence, Stage 2 requires one intermediate connection, and Stage 3 requires two connected evidence links. Baselines include an LLM without retrieval, dense RAG (Karpukhin et al., 2020), GraphRAG (Edge et al., 2025), LightRAG (Guo et al., 2025), HippoRAG 2 (Gutiérrez et al., 2025), and Hyper-RAG (Feng et al., 2026). Within each comparison and evaluation run, all methods share the documents, questions, reference evidence, generator, and evaluation protocol. GPT-4o-mini generates questions and evaluates answers against reference evidence according to six criteria, with scores ranging from 0 to 100. Five independent evaluation runs are conducted. In each run, we randomly generate a distinct 50-question evaluation set for each corpus. The mean $\pm$ standard deviation across the five run-level mean scores are reported; the average score is the mean across the six criteria. Appendix H reports the judging prompt.Appendix I provides corpus statistics, preprocessing, and the detailed evaluation protocol. Appendices K and L report the expert blind evaluation and evidence-chain retrieval analysis, respectively.

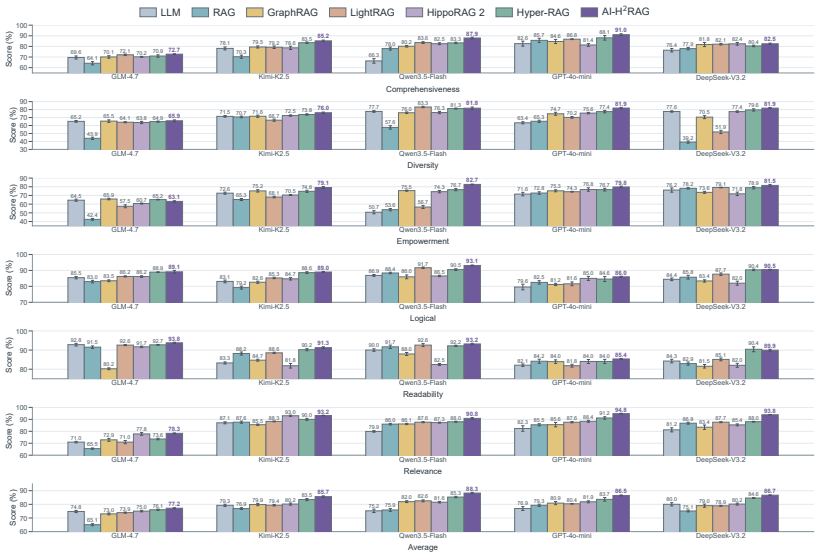


Figure 5: Six evaluation-dimension scores and average scores of different RAG methods under different LLM backbones.

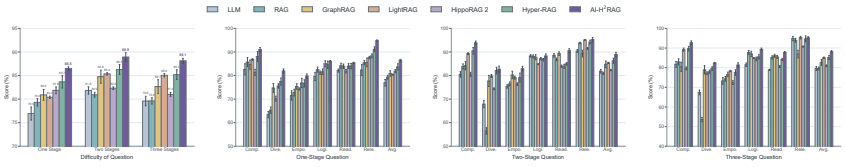


Figure 6: Six evaluation-dimension scores and average scores of different RAG methods across MechanicalCorp question stages.

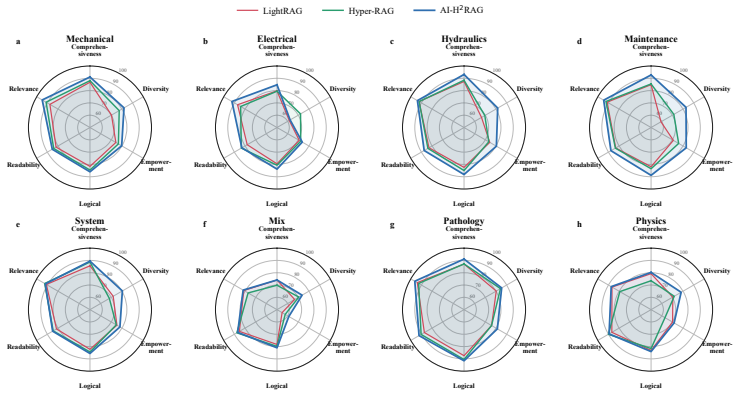


Figure 7: Six-dimensional cross-domain comparison of LightRAG, Hyper-RAG, and AI-H^{2}RAG across eight corpora under the same GPT-4o-mini backbone.

4.2 Overall Performance Across RAG Methods

At Stage 1 on MechanicalCorp, Figure 5 compares AI-H^{2}RAG with other RAG methods across LLM backbones: GLM-4.7, Kimi-K2.5, Qwen3.5-Flash, GPT-4o-mini, and DeepSeek-V3.2. AI-H^{2}RAG ranks first under every backbone, with average scores of 77.2, 85.7, 88.3, 86.5, and 86.7. Its relative improvement over the strongest baseline is 1.4–3.4% across the five backbones. Scores vary with the generator, from 77.2 under GLM-4.7 to 88.3 under Qwen3.5-Flash, but the rank ordering remains unchanged. Thus, the retrieval gain persists across generators with different answer capabilities.

MechanicalCorp provides a controlled comparison across evidence depths; stage details appear in Appendix Table J.1. With the generator fixed, AI-H^{2}RAG scores 86.5, 88.9, and 88.1 on Stages 1–3, yielding relative improvements of 3.3%, 3.0%, and 3.4%, respectively, over the strongest baseline, Hyper-RAG, as shown in Figure 6. The gains therefore persist across direct and multi-hop questions.

4.3 Cross-domain Generalization

We fix the generator and compare LightRAG, Hyper-RAG, and AI-H^{2}RAG across all eight corpora. Figures 7 and 8 show that AI-H^{2}RAG still achieves the highest average score in every domain.

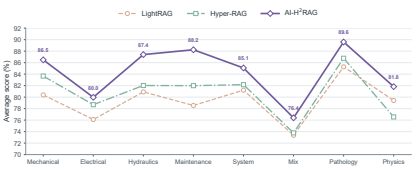


Figure 8: Cross-domain average scores under the same GPT-4o-mini backbone.

Its relative improvements over the strongest baseline are 3.3%, 1.7%, 6.6%, 7.6%, and 3.5% on the five machine-tool corpora, and 3.5%, 3.2%, and 3.0% on MixCorp, PathologyCorp, and PhysicsCorp. The radar profiles show improvements across most of the six criteria, rather than gains driven by a single dimension. All methods score lower on MixCorp, which contains more heterogeneous terminology and evidence patterns, yet AI-H^{2}RAG still ranks first. These results support transfer across both specialized technical and general-domain corpora.

4.4 Ablation Study

Four ablations are evaluated on 50 Stage 1 questions sampled from MechanicalCorp per evaluation run: hierarchical hypergraph memory, asymmetric hypergraph network, partial-cue query alignment, and coarsen-to-fine evidence retrieval. For hierarchical hypergraph memory, we replace the nested hypergraphs and membership mappings with a flat dense chunk index queried by the raw expanded-query representation. For asymmetric hypergraph network, we retain the hierarchy but use raw document representations without cross-level propagation. For partial-cue query alignment, we replace all aligned query representations with the raw expanded-query and raw query-entity representations. For coarsen-to-fine evidence retrieval, we bypass topic-scoped recall, sentence reranking, and entity-mediated expansion, and globally retrieve the top-B chunks using $\mathbf{h}_{q,\mathrm{C}}^{(\mathrm{final})\top}\mathbf{h}_{c_i}^{(\mathrm{final})}$.

Table 1: Ablations on 50 Stage 1 questions per evaluation run using GPT-4o-mini. Metrics are mean $\pm$ SD across five independent means; Δ is the relative mean-score change from the full model.

Ablated design	Comp.	Div.	Emp.	Log.	Read.	Rel.	Avg.	Δ (%)
Full Structure (AI-H^{2}RAG)	91.04 ± 0.83	81.94 ± 0.85	79.76 ± 0.91	86.00 ± 0.37	85.36 ± 0.27	94.80 ± 0.32	86.48 ± 0.38	0.0%
w/o Hierarchical Hypergraph Memory	83.80 ± 1.21	71.90 ± 1.05	75.70 ± 0.73	83.60 ± 0.44	84.10 ± 0.31	86.61 ± 0.29	80.95 ± 0.42	-6.4%
w/o Asymmetric Hypergraph Network	86.00 ± 1.24	71.70 ± 1.37	75.70 ± 0.84	85.50 ± 0.21	83.20 ± 0.44	92.40 ± 0.23	82.42 ± 0.58	-4.7%
w/o Partial-Cue Query Alignment	84.80 ± 1.03	75.70 ± 1.44	76.30 ± 0.65	84.60 ± 0.34	83.42 ± 0.38	91.54 ± 0.21	82.73 ± 0.38	-4.3%
w/o Coarse-to-Fine Evidence Retrieval	84.20 ± 0.97	69.40 ± 1.39	75.50 ± 0.78	83.90 ± 0.33	84.72 ± 0.37	89.20 ± 0.25	81.15 ± 0.54	-6.2%

Table 1 shows that all the ablated variants underperform the full model. Removing hierarchical hypergraph memory yields the largest average-score decrease of 6.4%, while bypassing coarsen-to-fine evidence retrieva reduces the score by 6.2%. Removing asymmetric hypergraph network and query alignment decreases the average score by 4.7% and 4.3%, respectively. These consistent declines indicate that hierarchical hypergraph construction, asymmetric indexing, query alignment, and hierarchical evidence retrieval each contribute to the overall performance of AI-H^{2}RAG.

4.5 Retrieval Efficiency

Furthermore, We measure online retrieval and context-construction time on 50 Stage 1 questions sampled from MechanicalCorp per evaluation run, excluding offline index construction and final answer generation. Each point in Figure 9 combines the mean latency of a retrieval method with its average answer-quality score. RAG and LightRAG are relatively responsive because their retrieval procedures involve less hierarchical structural processing. Although Hyper-RAG and AI-H^{2}RAG spend more time on structured retrieval, they achieve substantially higher answer-quality scores. Notably, AI-H^{2}RAG contains faster response than Hyper-RAG, while achieves the highest answer-quality score among the four approaches. Therefore, query alignment and coarsen-to-fine evidence retrieval therefore improve evidence organization without imposing a large latency increase over the strongest structured-retrieval baseline.

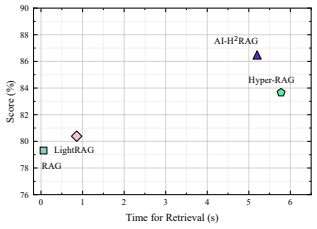


Figure 9: Average retrieval efficiency on 50 Stage 1 questions.

5 Conclusion

In this paper, we introduced AI-H^{2}RAG, a bio-inspired framework that treats document retrieval as coupled processes of hierarchical memory formation and cue-activated recall. It binds multi-level knowledge in nested hypergraphs, constructs an asymmetric retrieval network with raw evidence flowing upward and contextual guidance downward, and reuses the same hierarchy for coarse-to-fine evidence recovery. Evaluations span eight document corpora, five generator backbones, and three evidence-depth stages on MechanicalCorp. AI-H^{2}RAG achieves the highest average answer quality among the evaluated methods while maintaining lower latency compared to the strongest structured-retrieval baseline. These findings highlight the practical value of AI-H^{2}RAG for accurate and efficient question answering across diverse document settings.

AI USE STATEMENT

In this work, we used GPT-based generative AI tools to assist with code development and literature organization. We also used GPT-4o-mini to generate synthetic evaluation questions from sampled corpus chunks and to automatically evaluate generated answers against the corresponding reference evidence using a predefined six-criterion rubric. These uses were part of the experimental protocol described in the Experimental Setup and Appendix. All AI-assisted code, literature materials, and experimental outputs were reviewed by the authors. The authors take responsibility for the final manuscript, experimental design, reported results, and all claims in this submission.

ETHICS STATEMENT

AI-H^{2}RAG is designed for document-grounded question answering and is not intended to establish the truth or safety of retrieved information. Its outputs may reflect inaccuracies, biases, outdated knowledge, or sensitive content present in the underlying documents and the language models used for generation. It should therefore not be used as the sole basis for high-stakes decisions. We use automated LLM-based evaluation together with a blinded expert assessment of anonymized system outputs; no individual-level claims are drawn from these assessments.

REPRODUCIBILITY STATEMENT

To facilitate reproduction of our results, we document all implementation details and experimental configurations. Our code is provided as supplementary material and will be released on GitHub. We provide the complete formulation of hierarchical hypergraph memory, asymmetric hypergraph network, and coarsen-to-fine evidence retrieval. The appendix specifies fuzzy-topic construction, layer-wise update equations, extraction and evaluation prompts, corpus statistics, preprocessing procedures, baseline settings, and the LLM-judge rubric. We report results over five independent evaluation runs, with shared documents, questions, reference evidence, generators, and evaluation protocols for all compared methods. These details are intended to support reproduction of the reported experiments using the same corpora and model APIs.

REFERENCES

Omar Adjali, Olivier Ferret, Sahar Ghannay, and Hervé Le Borgne. Multi-level information retrieval augmented generation for knowledge-based visual question answering. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing*, pp. 16499–16513, Miami, Florida, USA, 2024. Association for Computational Linguistics. doi: 10.18653/v1/2024.emnlp-main.922. URL https://aclanthology.org/2024.emnlp-main.922/.

Akari Asai, Zeqi 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.

Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George Bm Van Den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, Diego De Las Casas, Aurelia Guy, Jacob Menick, Roman Ring, Tom Hennigan, Saffron Huang, Loren Maggiore, Chris Jones, Albin Cassirer, Andy Brock, Michela Paganini, Geoffrey Irving, Oriol Vinyals, Simon Osindero, Karen Simonyan, Jack Rae, Erich Elsen, and Laurent Sifre. Improving language models by retrieving from trillions of tokens. In *Proceedings of the 39th International Conference on Machine Learning*, volume 162 of *Proceedings of Machine Learning Research*, pp. 2206–2240, 2022. URL https://proceedings.mlr.press/v162/borgeaud22a.html.

Jianlyu Chen, Shitao Xiao, Peitian Zhang, Kun Luo, Defu Lian, and Zheng Liu. M3-Embedding: Multi-linguality, multi-functionality, multi-granularity text embeddings through self-knowledge distillation. In *Findings of the Association for Computational Linguistics: ACL 2024*, pp. 2318–2335, 2024. doi: 10.18653/v1/2024.findings-acl.137. URL https://aclanthology.org/2024.findings-acl.137/.

Eli Chien, Chao Pan, Jianhao Peng, and Olgica Milenkovic. You are AllSet: A multiset function framework for hypergraph neural networks. In *Proceedings of the Tenth International Conference on Learning Representations*, 2022.

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 GraphRAG approach to query-focused summarization. *arXiv preprint arXiv:2404.16130*, 2025.

Shahul Es, Jithin James, Luis Espinosa-Anke, and Steven Schockaert. RAGAs: Automated evaluation of retrieval augmented generation. In *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: System Demonstrations*, pp. 150–158, 2024. doi: 10.18653/v1/2024.eacl-demo.16.

Yifan Feng, Haoxuan You, Zizhao Zhang, Rongrong Ji, and Yue Gao. Hypergraph neural networks. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pp. 3558–3565, 2019. doi: 10.1609/aaai.v33i01.33013558.

Yifan Feng, Hao Hu, Shihui Ying, Xingliang Hou, Shiquan Liu, Mingyuan Yang, Junchang Li, Shaoyi Du, Nanning Zheng, Han Hu, and Yue Gao. Hyper-RAG: Combating LLM hallucinations using hypergraph-driven retrieval-augmented generation. *Nature Communications*, 17:5778, 2026. doi: 10.1038/s41467-026-71411-1. URL https://doi.org/10.1038/s41467-026-71411-1.

Yue Gao, Yifan Feng, Shuyi Ji, and Rongrong Ji. HGNN+: General hypergraph neural networks. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 45(3):3181–3199, 2023. doi: 10.1109/TPAMI.2022.3182052.

Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, and Haofen Wang. Retrieval-augmented generation for large language models: A survey. *arXiv preprint arXiv:2312.10997*, 2024.

Zirui Guo, Lianghao Xia, Yanhua Yu, Tu Ao, and Chao Huang. LightRAG: Simple and fast retrieval-augmented generation. *arXiv preprint arXiv:2410.05779*, 2025.

Bernal Jiménez Gutiérrez, Yiheng Shu, Weijian Qi, Sizhe Zhou, and Yu Su. From RAG to memory: Non-parametric continual learning for large language models. *arXiv preprint arXiv:2502.14802*, 2025. URL https://arxiv.org/abs/2502.14802.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. REALM: Retrieval-augmented language model pre-training. In *Proceedings of the 37th International Conference on Machine Learning*, volume 119 of *Proceedings of Machine Learning Research*, pp. 3929–3938, 2020.

William L. Hamilton, Rex Ying, and Jure Leskovec. Inductive representation learning on large graphs. In *Advances in Neural Information Processing Systems*, volume 30, pp. 1024–1034, 2017.

Johannes Hoffart, Mohamed Amir Yosef, Ilaria Bordino, Hagen Fürstenau, Manfred Pinkal, Marc Spaniol, Bilyana Taneva, Stefan Thater, and Gerhard Weikum. Robust disambiguation of named entities in text. In *Proceedings of the 2011 Conference on Empirical Methods in Natural Language Processing*, pp. 782–792, 2011. URL https://aclanthology.org/D11-1072/.

Yizheng Huang and Jimmy Xiangji Huang. A survey on retrieval-augmented text generation for large language models. *ACM Computing Surveys*, 58(12):1–38, 2026. doi: 10.1145/3805774.

Gautier Izacard, Patrick Lewis, Maria Lomeli, Lucas Hosseini, Fabio Petroni, Timo Schick, Jane Dwivedi-Yu, Armand Joulin, Sebastian Riedel, and Edouard Grave. Atlas: Few-shot learning with retrieval augmented language models. *Journal of Machine Learning Research*, 24(251): 1–43, 2023.

Bowen Jin, Gang Liu, Chi Han, Meng Jiang, Heng Ji, and Jiawei Han. Large language models on graphs: A comprehensive survey. *IEEE Transactions on Knowledge and Data Engineering*, 36 (12):8622–8642, 2024. doi: 10.1109/TKDE.2024.3469578.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pp. 6769–6781, 2020. doi: 10.18653/v1/2020.emnlp-main.550.

Thomas N. Kipf and Max Welling. Semi-supervised classification with graph convolutional networks. In *Proceedings of the Fifth International Conference on Learning Representations*, 2017.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Jacob Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. Natural questions: A benchmark for question answering research. *Transactions of the Association for Computational Linguistics*, 7:452–466, 2019. doi: 10.1162/tacl_a_00276.

Yunshi Lan, Gaole He, Jinhao Jiang, Jing Jiang, Wayne Xin Zhao, and Ji-Rong Wen. Complex knowledge base question answering: A survey. *IEEE Transactions on Knowledge and Data Engineering*, 35(11):11196–11215, 2023. doi: 10.1109/TKDE.2022.3223858.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Kuttler, Mike Lewis, Wen-tau Yih, Tim Rocktaschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems*, volume 33, pp. 9459–9474, 2020.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173, 2024. doi: 10.1162/tacl_a_00638.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. G-eval: NLG evaluation using GPT-4 with better human alignment. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 2511–2522, 2023. doi: 10.18653/v1/2023.emnlp-main.153.

Xinbei Ma, Yeyun Gong, Pengcheng He, Hai Zhao, and Nan Duan. Query rewriting in retrieval-augmented large language models. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pp. 5303–5315, 2023. doi: 10.18653/v1/2023.emnlp-main.322.

Shirui Pan, Linhao Luo, Yufei Wang, Chen Chen, Jiapu Wang, and Xindong Wu. Unifying large language models and knowledge graphs: A roadmap. *IEEE Transactions on Knowledge and Data Engineering*, 36(7):3580–3599, 2024. doi: 10.1109/TKDE.2024.3352100.

Boci Peng, Yun Zhu, Yongchao Liu, Xiaohe Bo, Haizhou Shi, Chuntao Hong, Yan Zhang, and Siliang Tang. Graph retrieval-augmented generation: A survey. *ACM Transactions on Information Systems*, 44(2):1–52, 2025. doi: 10.1145/3777378.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pp. 2383–2392, 2016. doi: 10.18653/v1/D16-1264.

Özge Sevgili, Artem Shelmanov, Mikhail Arkhipov, Alexander Panchenko, and Chris Biemann. Neural entity linking: A survey of models based on deep learning. *Semantic Web*, 13(3):527–570, 2022. doi: 10.3233/SW-222986.

Vincent A. Traag, Ludo Waltman, and Nees Jan van Eck. From louvain to leiden: Guaranteeing well-connected communities. *Scientific Reports*, 9(1):5233, 2019. doi: 10.1038/s41598-019-41695-z.

Petar Veličković, Guillem Cucurull, Arantxa Casanova, Adriana Romero, Pietro Liò, and Yoshua Bengio. Graph attention networks. In *Proceedings of the Sixth International Conference on Learning Representations*, 2018.

Zonghan Wu, Shirui Pan, Fengwen Chen, Guodong Long, Chengqi Zhang, and Philip S. Yu. A comprehensive survey on graph neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, 32(1):4–24, 2021. doi: 10.1109/TNNLS.2020.2978386.

Zhentao Xu, Mark Jerome Cruz, Matthew Guevara, Tie Wang, Manasi Deshpande, Xiaofeng Wang, and Zheng Li. Retrieval-augmented generation with knowledge graphs for customer service question answering. In *Proceedings of the 47th International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 2905–2909, 2024. doi: 10.1145/3626772.3661370.

Ikuya Yamada, Akari Asai, Hiroyuki Shindo, Hideaki Takeda, and Yuji Matsumoto. LUKE: Deep contextualized entity representations with entity-aware self-attention. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pp. 6442–6454, 2020. doi: 10.18653/v1/2020.emnlp-main.523.

Wenli Yang, Lilian Some, Michael Bain, and Byeong Kang. A comprehensive survey on integrating large language models with knowledge-based methods. *Knowledge-Based Systems*, 318:113503, 2025. doi: 10.1016/j.knosys.2025.113503.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric P. Xing, Hao Zhang, Joseph E. Gonzalez, and Ion Stoica. Judging LLM-as-a-judge with MT-Bench and chatbot arena. In *Advances in Neural Information Processing Systems*, volume 36, 2023.

Lingfeng Zhong, Jia Wu, Qian Li, Hao Peng, and Xindong Wu. A comprehensive survey on automatic knowledge graph construction. *ACM Computing Surveys*, 56(4):1–62, 2023. doi: 10.1145/3618295.

A Sentence-level Semantic-Unit Extraction Prompt

The following prompt is used to extract the sentence-level semantic units from the chunks.

```
SENTENCE_EXTRACTION_PROMPT = """
Extract all self-contained semantic units from the following paragraph.
    A semantic unit may contain one or more consecutive sentences when
    they together express one complete idea. Do not split sentences that
    depend on each other for essential context, and preserve the original
     wording.
Paragraph: {paragraph}
Return JSON only: {{"sentences": [{"text": "..."}, ...]}}
"""
```

B Entity Extraction Prompt

The following prompt is used to extract the entity-level semantic units from the sentences.

```
ENTITY_EXTRACTION_PROMPT = """
Extract the important information units from the sentence below.
    Identify both concrete entities and abstract semantic units that are
    useful for retrieval.

Sentence: {sentence}

Extract these types of information when they are explicitly present:
1. Concrete entities: people, organizations, locations, documents,
    products, methods, events, dates, and numerical values.
2. Abstract semantic units: central actions, goals, outcomes, causes,
    conditions, mechanisms, evidence, and query intent.

Importance scoring:
- 1.0: the main subject, core action, primary query intent, or
    indispensable result.
- 0.7-0.9: essential objects, key conditions, named entities, methods,
    evidence, or constraints.
- 0.4-0.6: useful modifiers, supporting details, and contextual
    attributes.
- 0.1-0.3: only include minor background details when they materially
    clarify the sentence.

Rules:
- Extract only text spans that occur in the sentence; do not invent
    entities or facts.
- Use concise, consistent type labels such as Person, Organization,
    Location, Document, Product, Event, Action, Goal, Method, Evidence,
    Condition, or QueryIntent.
- Assign a short description of each unit's role in the sentence.
```

```
Example 1 :
Input: "The Intervale supports new farmers through its Farms Program,
    which leases land, equipment, and storage facilities to commercial
    farmers. It also provides technical assistance and mentorship from
    experienced farmers."
Output:
{{"entities": [
  {{"name": "support new farmers", "type": "Action", "importance": 1.0, "
      description": "The central support action"}},
  {{"name": "Intervale", "type": "Organization", "importance": 0.9, "
      description": "The organization providing support"}},
  {{"name": "Farms Program", "type": "Program", "importance": 0.8, "
      description": "The support program"}},
  {{"name": "technical assistance and mentorship", "type": "Method", "
      importance": 0.8, "description": "Support mechanisms for new
      farmers"}}
]}}

Example 2 :
Input: "CytoSorb has received FDA Breakthrough Designation for the
    removal of ticagrelor in a cardiopulmonary bypass circuit during
    emergent and urgent cardiothoracic surgery. It also received
    Emergency Use Authorization for use in critically ill patients with
    COVID-19 infection."
Output:
{{"entities": [
  {{"name": "CytoSorb", "type": "Product", "importance": 1.0, "
      description": "The product being evaluated"}},
  {{"name": "FDA Breakthrough Designation", "type": "RegulatoryApproval",
       "importance": 1.0, "description": "A key regulatory milestone"}},
  {{"name": "Emergency Use Authorization", "type": "RegulatoryApproval",
      "importance": 1.0, "description": "An emergency regulatory
      authorization"}},
  {{"name": "ticagrelor", "type": "Drug", "importance": 0.8, "description
      ": "The substance targeted for removal"}},
  {{"name": "COVID-19 infection", "type": "Condition", "importance": 0.8,
       "description": "The condition of the authorized patient group"}}
]}}

Return exactly one JSON object and no additional text:
{{"entities": [...]}}
"""
```

C Latent Topic Construction from Chunk Similarity

This appendix details how the latent topic anchors $\mathcal{T} = \{t_k\}_{k=1}^{K}$ and the fuzzy topic–chunk membership matrix $\mathbf{M}^{\mathrm{T}\to\mathrm{C}} = [m_{ki}^{\mathrm{T}\to\mathrm{C}}] \in [0,1]^{K\times N}$ are induced from the chunk set $\mathcal{C} = \{c_i\}_{i=1}^{N}$. As defined in Equation 2, $\mathbf{h}_{c_i}^{(\mathrm{raw})} \in \mathbb{R}^d$ denotes the ℓ_2-normalized raw representation of chunk c_i. The pairwise semantic similarity between chunks is therefore computed by

$$s_{ij} = \mathbf{h}_{c_i}^{(\mathrm{raw})\top} \mathbf{h}_{c_j}^{(\mathrm{raw})}, \quad i \neq j. \tag{C.1}$$

To retain only informative semantic connections, we compute a dynamic threshold from the upper-triangular similarity values:

$$\tau = \mathrm{mean}\left(\{s_{ij} \mid i < j\}\right) + 0.5\,\mathrm{std}\left(\{s_{ij} \mid i < j\}\right). \tag{C.2}$$

We then construct a weighted undirected chunk-similarity graph $\mathcal{G} = (\mathcal{C}, \mathcal{R})$, where

$$(c_i, c_j) \in \mathcal{R} \iff s_{ij} > \tau, \qquad w_{ij} = s_{ij}. \tag{C.3}$$

Leiden community detection is applied to $\mathcal{G}$ with edge weights w_{ij} and resolution parameter 1.0, producing a partition $\{\mathcal{C}_k\}_{k=1}^{K}$ of the chunks (Traag et al., 2019). Each community $\mathcal{C}_k$ induces one latent topic t_k; together, these topics form $\mathcal{T}$. The representation of topic anchor t_k is the normalized centroid of its member chunk representations:

$$\boldsymbol{\mu}_k = \frac{1}{|\mathcal{C}_k|} \sum_{c_i \in \mathcal{C}_k} \mathbf{h}_{c_i}^{(\mathrm{raw})}, \qquad \mathbf{t}_k = \frac{\boldsymbol{\mu}_k}{|\boldsymbol{\mu}_k|_2}. \tag{C.4}$$

Although Leiden provides a primary topic assignment, we make the topic–chunk memberships fuzzy to preserve cross-topic associations. Given a fuzzy-membership threshold η, we define

$$m_{ki}^{\mathrm{T}\to\mathrm{C}} = \begin{cases} 1, & c_i \in \mathcal{C}_k, \\ {\mathbf{h}_{c_i}^{(\mathrm{raw})}}^{\top} \mathbf{t}_k, & c_i \notin \mathcal{C}_k \text{ and } {\mathbf{h}_{c_i}^{(\mathrm{raw})}}^{\top} \mathbf{t}_k > \eta, \\ 0, & \text{otherwise.} \end{cases} \tag{C.5}$$

Collecting these memberships yields $\mathbf{M}^{\mathrm{T}\to\mathrm{C}} = [m_{ki}^{\mathrm{T}\to\mathrm{C}}] \in [0,1]^{K\times N}$. Thus, each chunk is assigned membership one in its primary Leiden community. It can additionally belong to other topic hyperedges when its similarity to their topic anchors exceeds the threshold η. The topic-induced hyperedge anchored by t_k is

$$e_k^{\mathrm{T}\to\mathrm{C}} = \left\{ c_i \in \mathcal{C} \mid m_{ki}^{\mathrm{T}\to\mathrm{C}} > 0 \right\}. \tag{C.6}$$

The resulting hyperedge set $\mathcal{E}^{\mathrm{T}\to\mathrm{C}} = \{e_k^{\mathrm{T}\to\mathrm{C}}\}_{k=1}^{K}$ therefore defines the topic–chunk hypergraph $\mathcal{H}^{\mathrm{T}\to\mathrm{C}} = (\mathcal{C}, \mathcal{E}^{\mathrm{T}\to\mathrm{C}})$. A chunk may participate in multiple topic hyperedges when it has sufficiently high similarity to multiple topic anchors.

D Layer-wise Retrieval Network Updates

This appendix instantiates the asymmetric propagation rule in Equation 5 for the four document levels. All aggregation weights are inherited from the cross-level membership mappings defined in Section 3.1. The seed topic-anchor representations used for fuzzy-topic induction are not used during indexing; instead, indexing starts from the initialized topic representations $\mathbf{h}_{t_k}^{(\mathrm{raw})}$.

Topic Level. Topics form the root level and therefore receive no parent context. Their raw representations are retained as the final index representations:

$$\mathbf{h}_{t_k}^{(\mathrm{final})} = \mathbf{h}_{t_k}^{(\mathrm{raw})}, \qquad k = 1, \ldots, K. \tag{D.1}$$

Chunk Level. For each chunk c_i, final topic representations provide contextual guidance according to the fuzzy topic–chunk memberships, whereas raw sentence representations provide factual support according to chunk–sentence containment:

$$\mathbf{h}_{\mathrm{par}}^{(\mathrm{final})}(c_i) = \mathrm{norm}\left(\frac{\sum_{k=1}^{K} m_{ki}^{\mathrm{T}\to\mathrm{C}} \mathbf{h}_{t_k}^{(\mathrm{final})}}{\sum_{k=1}^{K} m_{ki}^{\mathrm{T}\to\mathrm{C}}} \right), \qquad \mathbf{h}_{\mathrm{child}}^{(\mathrm{raw})}(c_i) = \mathrm{norm}\left(\frac{\sum_{j=1}^{J} m_{ij}^{\mathrm{C}\to\mathrm{S}} \mathbf{h}_{s_j}^{(\mathrm{raw})}}{\sum_{j=1}^{J} m_{ij}^{\mathrm{C}\to\mathrm{S}}} \right). \tag{D.2}$$

Since $\mathbf{M}^{\mathrm{C}\to\mathrm{S}}$ is binary, the second term is the normalized mean of the raw sentence representations contained in c_i. The final chunk representation is

$$\mathbf{h}_{c_i}^{(\mathrm{final})} = \mathrm{Fuse}_{\mathrm{sim}}\left(\mathbf{h}_{c_i}^{(\mathrm{raw})}, \left\{ \mathbf{h}_{c_i}^{(\mathrm{raw})}, \mathbf{h}_{\mathrm{par}}^{(\mathrm{final})}(c_i), \mathbf{h}_{\mathrm{child}}^{(\mathrm{raw})}(c_i) \right\} \right). \tag{D.3}$$

Sentence Level. A sentence may be associated with multiple parent chunks when chunks overlap. Its parent context is therefore aggregated from final chunk representations, while its finer-level evidence is aggregated from raw semantic-unit representations:

$$\mathbf{h}_{\mathrm{par}}^{(\mathrm{final})}(s_j) = \mathrm{norm}\left(\frac{\sum_{i=1}^{N} m_{ij}^{\mathrm{C}\to\mathrm{S}} \mathbf{h}_{c_i}^{(\mathrm{final})}}{\sum_{i=1}^{N} m_{ij}^{\mathrm{C}\to\mathrm{S}}} \right), \qquad \mathbf{h}_{\mathrm{child}}^{(\mathrm{raw})}(s_j) = \mathrm{norm}\left(\frac{\sum_{m=1}^{M} m_{jm}^{\mathrm{S}\to\mathrm{U}} \mathbf{h}_{u_m}^{(\mathrm{raw})}}{\sum_{m=1}^{M} m_{jm}^{\mathrm{S}\to\mathrm{U}}} \right). \tag{D.4}$$

Because $m_{jm}^{\mathrm{S}\to\mathrm{U}}$ incorporates the importance score ρ_m, more important entities contribute more strongly to the raw factual support of s_j. The sentence representation is updated by

$$\mathbf{h}_{s_j}^{(\mathrm{final})} = \mathrm{Fuse}_{\mathrm{sim}}\left(\mathbf{h}_{s_j}^{(\mathrm{raw})}, \left\{\mathbf{h}_{s_j}^{(\mathrm{raw})}, \mathbf{h}_{\mathrm{par}}^{(\mathrm{final})}(s_j), \mathbf{h}_{\mathrm{child}}^{(\mathrm{raw})}(s_j)\right\}\right). \tag{D.5}$$

Entity Level. Entities form the finest document level. Each entity u_m therefore receives only final sentence context and retains its own raw representation:

$$\mathbf{h}_{\mathrm{par}}^{(\mathrm{final})}(u_m) = \mathrm{norm}\left(\frac{\sum_{j=1}^{J} m_{jm}^{\mathrm{S}\to\mathrm{U}} \mathbf{h}_{s_j}^{(\mathrm{final})}}{\sum_{j=1}^{J} m_{jm}^{\mathrm{S}\to\mathrm{U}}}\right), \tag{D.6}$$

$$\mathbf{h}_{u_m}^{(\mathrm{final})} = \mathrm{Fuse}_{\mathrm{sim}}\left(\mathbf{h}_{u_m}^{(\mathrm{raw})}, \left\{\mathbf{h}_{u_m}^{(\mathrm{raw})}, \mathbf{h}_{\mathrm{par}}^{(\mathrm{final})}(u_m)\right\}\right). \tag{D.7}$$

Here, we treat each u_m as a sentence-specific entity occurrence. It therefore receives parent context only from its source sentence, while identity links between occurrences sharing the same surface form are retained for cross-sentence expansion. The resulting dependency order is $\mathbf{h}_{t_k}^{(\mathrm{final})} \to \mathbf{h}_{c_i}^{(\mathrm{final})} \to \mathbf{h}_{s_j}^{(\mathrm{final})} \to \mathbf{h}_{u_m}^{(\mathrm{final})}$. At every update, contextual information is received only from final representations at the coarser level, whereas factual support is supplied only by raw representations at the finer level.

E Diagnostic for Same-Entity Different-Sense Disambiguation Across Document Contexts

To provide a targeted diagnostic of same-entity different-sense disambiguation across document contexts, we constructed a balanced set of 150 entity pairs from the fixed multi-domain corpus, comprising 75 same-sense pairs and 75 different-sense pairs. Each pair contains two occurrences of the same entity in different document contexts, and each occurrence is retained as a distinct document-side entity unit. We treated different-sense pairs as positive instances and used the cosine distance between their entity representations as the disambiguation score. We report AUROC with 95% confidence intervals obtained from 2,000 stratified bootstrap samples.

We compare a Raw condition, which uses the static embedding of each target entity, with an Asymmetric Indexing condition, which uses the final entity representation after hierarchical asymmetric indexing. As shown in Fig. E.1, Raw matching cannot distinguish the two classes because identical entities receive identical static embeddings. In contrast, the asymmetric-indexed entity representations achieve an AUROC of 0.774, with a 95% confidence interval of $[0.691, 0.854]$. Their paired AUROC improvement over Raw is 0.274, with a 95% bootstrap interval of $[0.191, 0.354]$. These results indicate that asymmetric indexing produces context-sensitive entity representations that distinguish different senses of the same entity in the evaluated corpus.

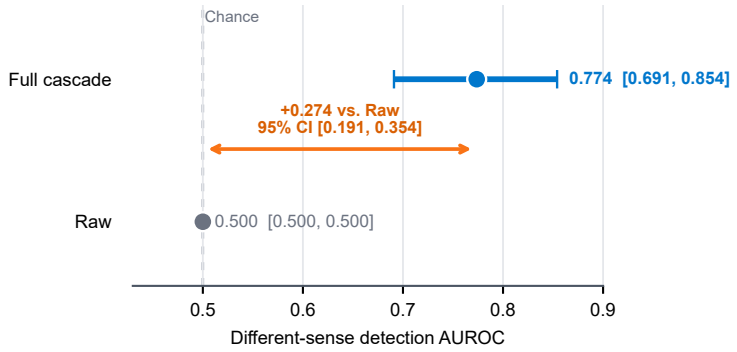


Figure E.1: Same-entity different-sense disambiguation diagnostic across document contexts. Different-sense entity pairs are treated as positive instances, and cosine distance is used as the disambiguation score. Points and horizontal bars show AUROC estimates and 95% confidence intervals, respectively. Higher AUROC is better.

F Query Representation Ablation for Sentence Reranking

This appendix evaluates the query representations used to compute the chunk-level and sentence-level relevance signals, $r_{j,\mathrm{C}}$ and $r_{j,\mathrm{S}}$, in Equation 12. We independently vary the representations used for the two signals between the raw expanded-query representation $\mathbf{h}_q^{(\mathrm{raw})}$ and the corresponding final query representations $\mathbf{h}_{q,\mathrm{C}}^{(\mathrm{final})}$ and $\mathbf{h}_{q,\mathrm{S}}^{(\mathrm{final})}$. All other retrieval and reranking components, including the entity-level signal $r_{j,\mathrm{U}}$, remain unchanged.

The main method uses $\mathbf{h}_q^{(\mathrm{raw})}$ for both $r_{j,\mathrm{C}}$ and $r_{j,\mathrm{S}}$. This choice retains the expanded query as a common matching anchor, while $\mathbf{h}_{q,\mathrm{C}}^{(\mathrm{final})}$ is used for topic-scoped chunk recall and $\mathbf{h}_{q,\mathrm{S}}^{(\mathrm{final})}$ contextualizes query entities for entity-level matching.

We evaluate 50 Stage 1 questions per run on MechanicalCorp using six metrics. Figure F.1 shows that using $\mathbf{h}_q^{(\mathrm{raw})}$ for both relevance signals achieves the highest mean score of 86.48. This result supports retaining the raw expanded query representation as the shared anchor for chunk-level and sentence-level matching during sentence reranking.

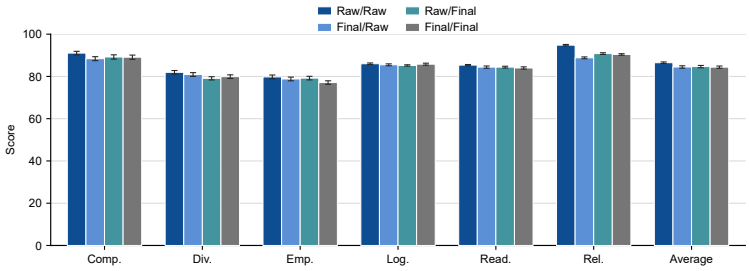


Figure F.1: Query-representation ablation for chunk-level and sentence-level matching scores during sentence reranking. c denotes $\mathbf{h}_q^{(\mathrm{raw})}$. Final denotes $\mathbf{h}_{q,\mathrm{C}}^{(\mathrm{final})}$ for chunk-level matching and $\mathbf{h}_{q,\mathrm{S}}^{(\mathrm{final})}$ for sentence-level matching. Error bars denote the standard deviation across the five run-level mean scores, with each run containing 50 Stage 1 questions. Higher is better for all metrics.

G Entity-Mediated Multi-Hop Evidence Expansion

This appendix details the entity-mediated expansion procedure used after sentence reranking in Section 3. The first-stage sentence evidence is obtained by selecting the highest-ranked sentences according to Equation 13, denoted by $\mathcal{R}_{\mathrm{S}}^{(1)}(q)$.

For each first-stage sentence $s_j \in \mathcal{R}_{\mathrm{S}}^{(1)}(q)$, we use its extracted entities $\mathcal{U}(s_j)$ to locate unvisited sentences that mention the same entity. Let $\mathrm{key}(u)$ denote the normalized textual form of entity u. The expansion candidate set is defined as

$$\mathcal{S}_{\mathrm{hop}}(q) = \left\{ s_{j'} \in \mathcal{S} \setminus \mathcal{R}_{\mathrm{S}}^{(1)}(q) \;\middle|\; \exists\, s_j \in \mathcal{R}_{\mathrm{S}}^{(1)}(q),\, u_m \in \mathcal{U}(s_j),\, u_{m'} \in \mathcal{U}(s_{j'}) : \mathrm{key}(u_m) = \mathrm{key}(u_{m'}) \right\}. \tag{G.1}$$

Each expanded candidate $s_{j'} \in \mathcal{S}_{\mathrm{hop}}(q)$ is scored using the same chunk-level, sentence-level, and entity-level relevance signals in Equations 12 and 13. To prioritize directly retrieved evidence, we apply a hop-decay factor $\lambda_{\mathrm{hop}} \in (0, 1]$:

$$R_{j'}^{(2)} = \lambda_{\mathrm{hop}} R_{j'}, \qquad s_{j'} \in \mathcal{S}_{\mathrm{hop}}(q), \tag{G.2}$$

where $R_{j'}$ is the reranking score before decay. The highest-ranked expanded sentences form the second-stage evidence $\mathcal{R}_{\mathrm{S}}^{(2)}(q)$. The final sentence evidence supplied to generation is

$$\mathcal{R}_{\mathrm{S}}(q) = \mathcal{R}_{\mathrm{S}}^{(1)}(q) \cup \mathcal{R}_{\mathrm{S}}^{(2)}(q). \tag{G.3}$$

For an n-hop expansion, the above procedure is applied iteratively, using the evidence obtained at the previous hop to construct and rerank candidates for the next hop. By linking sentences through

shared entities, the expansion can recover evidence located outside the chunks selected during coarse retrieval.

H LLM JUDGE SCORING PROMPT

This appendix provides the scoring guidelines for the LLM judge. For each evaluation instance, the question, generated answer, and corresponding reference evidence are provided together with this rubric. The judge assigns one level and one numerical score within the corresponding range for each criterion.

```
SCORING_PROMPT = """
You will evaluate the answers to the questions by using the relevant
    documents based on six criteria: **Comprehensiveness**, **Diversity
    **, **Empowerment**, **Logical**, **Readability**, and **Relevance**.

- **Comprehensiveness** -
Measure whether the answer comprehensively covers all key aspects of the
     question and whether there are omissions.
Level | score range | description
Level 1 | 0-20 | The answer is extremely one-sided, leaving out key
    parts or important aspects of the question.
Level 2 | 20-40 | The answer has some content, but it misses many
    important aspects of the question and is not comprehensive enough.
Level 3 | 40-60 | The answer is more comprehensive, covering the main
    aspects of the question, but there are still some omissions.
Level 4 | 60-80 | The answer is comprehensive, covering most aspects of
    the question, with few omissions.
Level 5 | 80-100 | The answer is extremely comprehensive, covering all
    aspects of the question with no omissions.

- **Diversity** -
Measure the richness of the answer content, including background
    knowledge, extended information, case studies, etc.
Level | score range | description
Level 1 | 0-20 | The answer is extremely sparse, providing only direct
    answers without additional information.
Level 2 | 20-40 | The answer provides a direct answer but contains only
    a small amount of relevant knowledge expansion.
Level 3 | 40-60 | In addition to the direct answers, the answer also
    provides some relevant background knowledge.
Level 4 | 60-80 | The answer is rich, providing more relevant background
     knowledge and supplementary information.
Level 5 | 80-100 | The answer provides a lot of relevant knowledge,
    expanded content and in-depth analysis.

- **Empowerment** -
Measure the credibility of the answer and whether it convinces the
    reader that it is correct.
Level | score range | description
Level 1 | 0-20 | The answer lacks credibility, contains obvious errors
    or false information.
Level 2 | 20-40 | The answer has some credibility, but some information
    is not accurate.
Level 3 | 40-60 | The answer is credible and provides some supporting
    information.
Level 4 | 60-80 | The answer is highly credible, providing sufficient
    supporting information.
Level 5 | 80-100 | The answer is highly credible with authoritative
    supporting information.

- **Logical** -
Measure whether the answers are coherent, clear, and easy to understand.
Level | score range | description
```

```
Level 1 | 0-20 | The answer is illogical, incoherent, and difficult to
    understand.
Level 2 | 20-40 | The answer has some logic, but is incoherent in parts.
Level 3 | 40-60 | The answer is logically clear and basically coherent.
Level 4 | 60-80 | The answer is logical, coherent, and easy to
    understand.
Level 5 | 80-100 | The answer is extremely logical, fluent and well-
    organized.

- **Readability** -
Measure whether the answer is well organized, clear in format, and easy
    to read.
Level | score range | description
Level 1 | 0-20 | The format is confused and difficult to read.
Level 2 | 20-40 | There are some problems in the format.
Level 3 | 40-60 | The format is basically clear.
Level 4 | 60-80 | The format is clear and well organized.
Level 5 | 80-100 | The format is very clear with excellent reading
    experience.

- **Relevance** -
Measure whether the reasoning and answer are highly relevant and helpful
     to the question.
Level | score range | description
Level 1 | 0-20 | The answer is entirely irrelevant, barely related to
    the question, or largely unhelpful.
Level 2 | 20-40 | The answer has limited relevance; much of the response
     is off-topic or unhelpful.
Level 3 | 40-60 | The answer is generally relevant, but includes
    distractions or less helpful parts.
Level 4 | 60-80 | The answer is mostly on point, with only minor
    digressions and overall useful content.
Level 5 | 80-100 | The answer is fully focused on the question, highly
    relevant, and highly helpful.

For each indicator, give a Level and a Score within the level's score
    range.
"""
```

I Detailed Experimental Setup

I.1 Evaluation Corpora

We use eight document corpora: five machine-tool corpora (MechanicalCorp, HydraulicsCorp, ElectricalCorp, MaintenanceCorp, and SystemCorp) and three general-domain corpora (MixCorp, PathologyCorp, and PhysicsCorp). The machine-tool corpora contain hierarchical technical descriptions and dense domain entities, whereas the three general-domain corpora test transfer under different terminology and document structures. Raw documents are deduplicated and converted into textual contexts. In each evaluation run, questions are generated from newly sampled corpus chunks, and their source contexts are retained as reference evidence.

MechanicalCorp is the controlled corpus for detailed analysis. In each evaluation run, it uses three 50-question stages: Stage 1 requires direct factual evidence, Stage 2 requires one intermediate evidence connection, and Stage 3 requires two connected evidence links. This progression tests retrieval from local support to compositional multi-hop evidence. Table I.1 reports corpus sizes and question counts per evaluation run.

Table I.1: Statistics of the evaluation corpora.

Corpus	Domain	Words	Questions / run
MechanicalCorp	Machine-tool	3,161,292	50 / 50 / 50
HydraulicsCorp	Machine-tool	2,145,104	50
ElectricalCorp	Machine-tool	2,189,922	50
MaintenanceCorp	Machine-tool	4,228,233	50
SystemCorp	Machine-tool	3,291,093	50
MixCorp	General-domain	2,196,607	50
PathologyCorp	General-domain	3,529,960	50
PhysicsCorp	General-domain	2,369,660	50

Word counts exclude whitespace after preprocessing and deduplication. The three MechanicalCorp question counts correspond to the three stages in each evaluation run. Within each run, all methods use the same documents, questions, reference evidence, generation model, and evaluation protocol. GPT-4o-mini is used for question generation and automatic answer evaluation.

I.2 Baselines

We compare with an LLM without retrieval, dense RAG (Karpukhin et al., 2020), GraphRAG (Edge et al., 2025), LightRAG (Guo et al., 2025), HippoRAG 2 (Gutiérrez et al., 2025), and Hyper-RAG (Feng et al., 2026). These baselines span unstructured retrieval, graph-based retrieval, memory-inspired graph retrieval, and hypergraph retrieval.

I.3 Evaluation Metrics

We use an LLM judge for open-ended knowledge-intensive QA (Liu et al., 2023; Zheng et al., 2023) and evaluate answers against the corresponding reference evidence. The six criteria are Comprehensiveness, Diversity, Empowerment, Logical, Readability, and Relevance, each scored from 0 to 100. All methods receive the same reference evidence, rubric, and judging prompt (Appendix H). We conduct five independent evaluation runs. In each run, we randomly sample different corpus chunks and generate a distinct 50-question evaluation set for each corpus and MechanicalCorp stage. We report the mean $\pm$ standard deviation across the five run-level mean scores. The average score is the mean over the six criteria.

J Representative MechanicalCorp Questions Across Evidence Depths

Table J.1 presents representative questions from the three evidence-depth stages of MechanicalCorp. Stage 1 requires direct factual support, Stage 2 introduces one intermediate evidence connection, and Stage 3 requires two connected evidence links. These examples illustrate how the evaluation progresses from retrieving local facts to assembling connected evidence for multi-hop answers.

Table J.1: Representative questions sampled from MechanicalCorp across the three stages.

Stage	Representative question	Evidence requirement
Stage 1	What factors influence the dynamic rigidity of spindle components in CNC machine tools?	Direct factual evidence.
Stage 2	What are the key design requirements for CNC machine tool spindles to improve their precision and rigidity, and how do these requirements impact the machining accuracy of the workpiece?	An intermediate connection involving the spindle design requirements and their effect on machining accuracy.
Stage 3	Explain how the design requirements for CNC machine tool spindles emphasize the need for high precision and rigidity, and specify the methods used to improve the rigidity of spindle components, particularly focusing on bearing configurations and preloading techniques.	Two connected evidence links involving spindle requirements, rigidity-improvement methods, and bearing design.

K EXPERT BLIND EVALUATION

Table K.1: Expert blind evaluation on 60 paired questions sampled from MechanicalCorp. Scores are question-level means over three experts on a 1–5 scale. Δ is AI-H^{2}RAG minus Hyper-RAG; CIs are 95% paired-bootstrap CIs.

Metric	Hyper-RAG	AI-H^{2}RAG	Δ (Casc–Hyper)	95% CI	ICC(A,1)
Correctness	4.694	**4.956**	+0.261	[0.083, 0.489]	0.757
Completeness	4.567	**4.806**	+0.239	[0.056, 0.456]	0.770
Clarity	4.672	**4.878**	+0.206	[0.089, 0.333]	0.765

To complement the LLM judge, we conduct a blinded comparison of AI-H^{2}RAG and Hyper-RAG on 60 questions randomly sampled from MechanicalCorp. Three machine-tool experts independently receive the question, reference answer, and two anonymous system answers. The mapping to System A or B and their presentation order are randomized for each expert and question, then revealed only after the forms are locked.

Experts score Correctness, Completeness, and Clarity on a 1–5 scale and select an overall preference. We average scores across experts for each question, estimate paired differences with 10,000 bootstrap resamples, and report 95% confidence intervals. Inter-rater agreement is measured by two-way random-effects, absolute-agreement ICC(A,1). Table K.1 shows higher scores for AI-H^{2}RAG in Correctness (4.956 vs. 4.694), Completeness (4.806 vs. 4.567), and Clarity (4.878 vs. 4.672). All paired-bootstrap intervals exclude zero. Of 180 paired judgments, experts preferred AI-H^{2}RAG in 51, Hyper-RAG in 26, and selected a tie in 103. This independent assessment is consistent with the automatic results.

L KEY-EVIDENCE COVERAGE AND MULTI-HOP EVIDENCE-CHAIN RETRIEVAL

Table L.1: Evidence coverage on the frozen evaluation set derived from MechanicalCorp. Values in the three recall columns are percentages, reported as AI-H^{2}RAG / Hyper-RAG. Δ is AI-H^{2}RAG minus Hyper-RAG in percentage points, with 95% paired-bootstrap confidence intervals.

Group	Sentence Recall	Bridge Recall	Full-chain Recall	Δ Full-chain Recall (95% CI)
Stage 1	**95.0** / 50.0	—	**95.0** / 50.0	**+45.0** [25.0, 65.0]
Stage 2	**86.0** / 67.0	**84.0** / 64.0	**84.0** / 64.0	**+20.0** [6.1, 34.7]
Stage 3	**87.3** / 69.3	**64.0** / 50.0	**58.0** / 46.0	**+12.0** [0.2, 24.0]

Sentence Recall is the proportion of annotated gold source sentences visible in the final source evidence. Bridge Recall requires an annotated bridge entity and both gold source sentences linked by that entity to be visible. Full-chain Recall requires all annotated gold source sentences and bridge entities needed to form the complete evidence chain to be visible.

We finally test whether the answer-quality gain is accompanied by better retrieval of generator-visible evidence. Two annotators construct a frozen 120-question subset of MechanicalCorp for gold-evidence evaluation, comprising 20 questions from Stage 1, 50 from Stage 2, and 50 from Stage 3. Disagreements are adjudicated before evaluation. We map the native outputs of AI-H^{2}RAG and Hyper-RAG to the same original-document sentence identifiers and evaluate both under a shared 12,000-token source-evidence budget. Table L.1 shows that AI-H^{2}RAG outperforms Hyper-RAG on all applicable metrics across the three stages. Full-chain recall increases from 50.0% to 95.0% in Stage 1, from 64.0% to 84.0% in Stage 2, and from 46.0% to 58.0% in Stage 3. These improvements correspond to gains of 45.0, 20.0, and 12.0 percentage points, respectively, with all paired-bootstrap 95% confidence intervals excluding zero. For Stages 2 and 3, bridge recall also improves by 20.0 and 14.0 percentage points, respectively. These results show that cascaded retrieval exposes both key source sentences and entity-mediated links more reliably to the generator.