ORIGINRAG: HARNESSING GRAPHRAG BY TRACING EVIDENCE ORIGINS

Anonymous authors
Paper under double-blind review

ABSTRACT

GraphRAG gathers supporting evidence for multi-hop questions from corpus graphs. However, prior methods do not explicitly expose where retrieved evidence originates and how retrieval decisions connect to source grounding. We propose OriginRAG, a novel framework that harnesses GraphRAG by maintaining a set of demand-specific provenance lineages as its retrieval state. It derives evidence demands from the query and initializes one provenance lineage per demand. Each lineage is expanded by demand-conditioned graph retrieval, so every candidate preserves its origin: the demand that triggered its retrieval. Such origin tracing helps an agent choose complementary passages while completing a lineage. Once a demand is satisfied, it is grounded in a verbatim span from the selected evidence. On three multi-hop QA benchmarks, OriginRAG improves over the mean of four GraphRAG baselines by an average of 5.59 points across R@5, EM, and F1.

1 INTRODUCTION

Retrieval-augmented generation (RAG) supplies a language model with evidence retrieved from an external corpus (Lewis et al., 2020). Conventional retrievers score passages independently, which works when one passage contains the required evidence. Multi-hop questions often depend on bridge passages whose relevance emerges through previously discovered evidence. Graph-based RAG (GraphRAG) addresses this dependency by organizing corpus units through structured relations. HippoRAG propagates relevance across an OpenIE graph using Personalized PageRank (Gutiérrez et al., 2024). HippoRAG 2 deepens passage integration through online language-model use (Gutiérrez et al., 2025). PropRAG retrieves proposition paths (Wang & Han, 2025). HGRAG uses a cross-granularity hypergraph (Wang et al., 2026a). Graph-based retrieval finds relevant candidates, but the final evidence set does not record the requirement behind each source. This missing connection can leave GraphRAG with an incomplete evidence chain. Figure 1 presents a representative GraphRAG failure on HotpotQA. The retrieved passages cover only part of the question, leaving one evidence demand unresolved. This example motivates our central question. ***How can GraphRAG carry each evidence demand from graph discovery to source grounding?***

Previous work uses query structure at different points in retrieval. Question decomposition preserves subqueries during candidate discovery (Ammann et al., 2025). SETR uses explicit information requirements during evidence selection (Lee et al., 2025b), while RichRAG retains aspect–document associations during ranking (Wang et al., 2025). Relink organizes retrieved evidence into a query-specific graph (Huang et al., 2026). NeocorRAG organizes retrieval around evidence chains (Peng et al., 2026). These methods improve coverage during discovery, ranking, or selection, but do not carry the same demand–passage association through graph retrieval to verbatim source evidence.

Lineage tracing offers a natural model for tracking evidence origins. In biology, it identifies the descendants of a marked cell (Kretzschmar & Watt, 2012). Data lineage applies the same principle to derived records by identifying the source items that produced them (Cui et al., 2000). For GraphRAG, we define a ***provenance lineage*** as the trace that carries one evidence demand from graph discovery to its verbatim source evidence.

We propose OriginRAG, a GraphRAG framework that maintains demand-specific provenance lineages as its retrieval state in three stages. ***Evidence-Demand Induction*** derives query-specific demands, each specifying what the retrieved evidence must establish. ***Demand-Conditioned Lineage Expansion*** retrieves candidates for each demand and preserves their demand associations during

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.

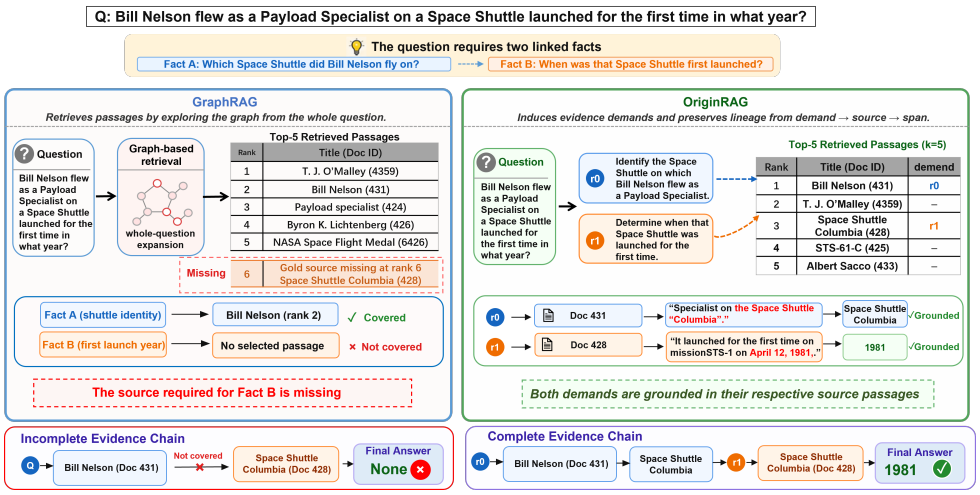


Figure 1: A GraphRAG failure case on HotpotQA under R@5. Whole-question ranking misses the Columbia passage required for the second hop and leaves the evidence chain incomplete. OriginRAG retrieves evidence for both hops using separate demands and returns 1981 as the correct answer.

candidate merging. ***Lineage Completion with Source Grounding*** uses this provenance to select complementary passages and extract source-anchored verbatim spans from the selected passages. Retaining such a span completes the corresponding lineage.

Our contributions are as follows.

- We formulate provenance lineage as a trace that follows each Evidence Demand from graph discovery to verbatim source evidence. It keeps the provenance of each demand explicit while the final passage set is constructed.
- We introduce OriginRAG, which preserves each Evidence Demand's identity from candidate discovery to grounded source evidence. This correspondence guides passage selection and links each supported demand to a source-anchored span.
- Experiments on multi-hop QA benchmarks show that OriginRAG achieves the strongest overall results, with average gains of 3.59 R@5, 6.81 EM, and 6.37 F1 points over the leading GraphRAG baselines.

2 Related Work

GraphRAG. GraphRAG represents corpus knowledge or query-specific evidence as a graph and retrieves through its relations. HippoRAG propagates relevance over an OpenIE graph (Gutiérrez et al., 2024), and HippoRAG 2 strengthens the integration of passages and graph retrieval (Gutiérrez et al., 2025). SubgraphRAG combines directional distance encoding with lightweight triple scoring to retrieve evidence subgraphs (Li et al., 2025b). PropRAG searches proposition paths with an online beam search (Wang & Han, 2025). LightRAG combines graph and vector retrieval at two granularities (Guo et al., 2025), while HGRAG retrieves over a cross-granularity hypergraph (Wang et al., 2026a). G-reasoner trains a graph foundation model over its QuadGraph abstraction (Luo et al., 2026). SUBQRAG stores triples from sub-question reasoning as graph memory (Li et al., 2025a). AGRAG presents a reasoning subgraph with the retrieved chunks (Wang et al., 2026b). These systems improve evidence discovery through graph structure. OriginRAG also keeps the connection between each Evidence Demand and its retrieved sources available during passage selection and source grounding.

Query decomposition and evidence selection. Question decomposition merges subquery retrieval results for reranking (Ammann et al., 2025). LogicRAG schedules subproblem retrieval in DAG topological order (Chen et al., 2026). DeCoR uses decomposed queries and compressed evidence for second-stage retrieval (Yun & Kim, 2025). S2G-RAG predicts evidence sufficiency and structured

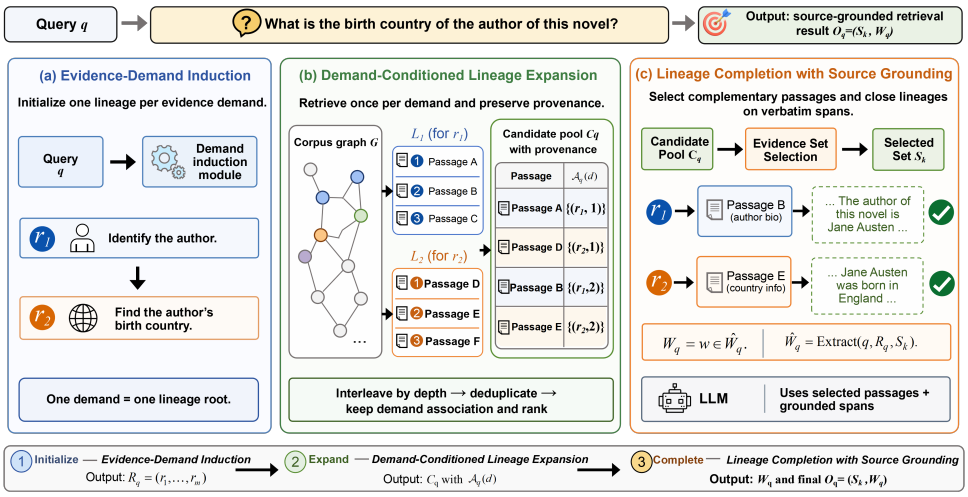


Figure 2: Overview of OriginRAG. **Evidence-Demand Induction** identifies facts required by the query. **Demand-Conditioned Lineage Expansion** retrieves candidates for each demand and preserves demand–passage provenance. **Lineage Completion with Source Grounding** selects complementary passages and grounds each supported demand in a verbatim source span.

gaps to guide iterative retrieval and sentence-level evidence extraction (Li et al., 2026). SETR selects evidence using explicit information requirements (Lee et al., 2025b), while RichRAG ranks with aspect–document associations (Wang et al., 2025). Relink constructs query-driven evidence graphs (Huang et al., 2026), NeocorRAG retrieves along evidence chains (Peng et al., 2026), and ETS performs tree search over evidence sets (Sun et al., 2025). OriginRAG preserves each Evidence Demand's identity through retrieval, selection, and grounding to a verbatim source span.

3 Problem Formulation

Let q be a query and $\mathcal{D} = \{d_i\}_{i=1}^{N}$ a passage corpus. For $d \in \mathcal{D}$, $\text{text}(d)$ denotes its source text. $\mathcal{G}$ is the graph index over this corpus. The parameter k specifies the context size. An *Evidence Demand* is a query-specific evidence requirement. The method produces

$$\mathcal{O}_q = (S_k, W_q), \qquad S_k = (d^{(1)}, \ldots, d^{(k)}) \in \mathcal{D}^k, \tag{1}$$

where S_k is the ordered list of selected passages and W_q is the set of grounded annotations. The passages in S_k are distinct. Each annotation $w = (i_w, d_w, s_w, a_w)$ links demand i_w to span s_w in selected passage d_w. The term a_w records the resulting structured relation.

4 OriginRAG

OriginRAG traces evidence origins through demand-specific provenance lineages from graph retrieval to source grounding in three stages. Figure 2 illustrates these stages with an example.

4.1 Evidence-Demand Induction

A language model uses a fixed prompt $\mathcal{P}_D$ to induce m Evidence Demands from q, subject to a per-query limit $m_{\max}$:

$$R_q = \text{Normalize}(\text{LLM}_\theta(\mathcal{P}_D, q)) = (r_1, \ldots, r_m). \tag{2}$$

$$r_j = (\text{id}_j, \delta_j, I_j, O_j), \qquad 1 \le j \le m \le m_{\max}. \tag{3}$$

The description δ_j states the fact that evidence must establish. I_j records the symbolic variables consumed by the demand, and O_j records those its evidence should establish. They are not values resolved from retrieved evidence. All demands are induced jointly from q. In Figure 1, r_0 identifies

the shuttle and r_1 requests its launch year. The identifier id_j remains fixed through retrieval, selection, and grounding.

The induction prompt asks the model to express the required facts as ordered demands:

Evidence-Demand Induction Prompt $\mathcal{P}_D$

Decompose the query into ordered Evidence Demands whose joint resolution makes the answer derivable. Each demand states one fact that a source passage must establish. Separate evidence identifying an intermediate entity from evidence establishing the answer.

The induction instructions are summarized in Appendix A.1.

4.2 Demand-Conditioned Lineage Expansion

For each Evidence Demand, OriginRAG retrieves a ranked passage list from the same graph index:

$$L_j = \mathrm{Retrieve}_{\mathcal{G}}(q, \delta_j) = (d_{j,1}, \ldots, d_{j,h}), \tag{4}$$

where h is the retrieval depth. The retrieval query formed from q and δ_j determines the graph seeds. Personalized PageRank propagates the seed scores over $\mathcal{G}$ and ranks passages to form L_j. Each call uses (q, δ_j) independently.

The order of R_q controls depth-wise list merging and demand presentation to selection and grounding. A passage enters $C_q = (c_1, \ldots, c_M)$ at its first occurrence, with $M \geq k$. Let $\tau(d)$ be its normalized title. The retained provenance and selector representation are

$$\mathcal{A}_q(d) = \{(\mathrm{id}_j, \ell) \mid d = d_{j,\ell}\}. \tag{5}$$

$$b_q(d) = (\mathcal{A}_q(d), n_q(d), \rho_q(d), g_q(d)), \qquad d \in C_q. \tag{6}$$

$\mathcal{A}_q(d)$ preserves each candidate's retrieval origins: the demand ID and rank of every retrieval event that exposed d. The count $n_q(d) = |\mathcal{A}_q(d)|$ summarizes repeated exposure. The statistic $\rho_q(d)$ is the best rank recorded in $\mathcal{A}_q(d)$. Same-title redundancy is represented by $g_q(d) = |\{c \in C_q : \tau(c) = \tau(d)\}|$.

During merging, OriginRAG retains each candidate's demand IDs and within-demand ranks in $\mathcal{A}_q(d)$. The selector can then distinguish candidates retrieved for different Evidence Demands and construct a passage set that covers complementary requirements.

4.3 Lineage Completion with Source Grounding

Evidence set selection. The selector evaluates the merged candidate pool against R_q. Each candidate excerpt is augmented by $b_q(d)$:

$$S_k = \mathrm{Select}_k(q, R_q, C_q, \{b_q(d)\}_{d \in C_q}). \tag{7}$$

The selector evaluates candidates jointly to build the passage set. The provenance in $b_q(d)$ reveals whether a candidate contributes to an uncovered demand. It can therefore favor a complementary passage even when a redundant passage is ranked higher in the merged pool.

The selector applies this provenance through the following instruction:

Evidence Set Selection Prompt

Use each candidate's demand-specific retrieval provenance to select k passages that jointly cover the Evidence Demands. Favor complementary facts and avoid evidence that repeats a covered demand.

Returned IDs are validated against C_q and deduplicated. The retained passages follow candidate-pool order, and the earliest unselected candidates fill any remaining positions up to k. The selector prompt appears in Appendix A.2.

Source grounding. After S_k is fixed, source grounding extracts demand-linked verbatim spans from the selected passages. Candidate annotations are proposed by

$$\widehat{W}_q = \mathrm{Extract}(q, R_q, S_k). \tag{8}$$

The joint grounding call uses I_j and O_j to populate the `consumes` and `produces` bindings. Provenance lineage preserves demand identity across stages. The extraction prompt is:

Source Grounding Prompt

For each supported Evidence Demand, return its source passage and the verbatim span that establishes it. Return no record for unsupported demands.

The proposed annotations are filtered by

$$W_q = \{w \in \widehat{W}_q : \mathrm{Valid}_q(w)\}. \tag{9}$$

Validity first anchors the proposal to a selected source: $d_w \in S_k$ and $s_w \sqsubseteq_{\mathrm{norm}} \mathrm{text}(d_w)$. Here $\sqsubseteq_{\mathrm{norm}}$ is substring containment after whitespace and case normalization. The schema check $V_\Gamma(a_w) = 1$ validates the structure of the annotation. Provenance continuity further requires $\exists \ell\,(i_w, \ell) \in \mathcal{A}_q(d_w)$. These conditions jointly define $\mathrm{Valid}_q(w)$.

Duplicate passage–span pairs are removed before presentation. The reader receives the retained annotations W_q alongside the selected passages S_k. A lineage remains open when W_q contains no valid annotation for its demand. The grounding and QA prompts appear in Appendices A.3 and A.4.

5 Experiments

We evaluate OriginRAG through four research questions:

- **RQ1.** How does OriginRAG compare with leading GraphRAG methods on multi-hop QA benchmarks?
- **RQ2.** How do the components of OriginRAG contribute to retrieval and answer quality?
- **RQ3.** How do evidence-set construction and source grounding account for OriginRAG's retrieval and QA gains?
- **RQ4.** What is the computational cost of OriginRAG?

5.1 Experimental Setup

Datasets. We evaluate on three multi-hop benchmarks: HotpotQA (Yang et al., 2018), 2WikiMultiHopQA (Ho et al., 2020), and MuSiQue (Trivedi et al., 2022). Following HippoRAG 2 (Gutiérrez et al., 2025), we use 1,000 queries from each dataset and its corresponding retrieval corpus.

Baselines. The baselines are grouped as follows: **(1) LLM-based retrieval: NV-Embed-v2** (Lee et al., 2025a). **(2) Conventional RAG: BM25** (Robertson & Zaragoza, 2009), **RAPTOR** (Sarthi et al., 2024), and **LogicRAG** (Chen et al., 2026). **(3) GraphRAG: HippoRAG 2** (Gutiérrez et al., 2025), **PropRAG** (Wang & Han, 2025), **HGRAG** (Wang et al., 2026a), and **G-reasoner** (Luo et al., 2026). **(4) Evidence selection methods: ETS-7B** (Sun et al., 2025), **Relink** (Huang et al., 2026), and **NeocorRAG** (Peng et al., 2026).

Metrics. Retrieval is evaluated with supporting-passage R@5, and QA with exact match (EM) and token-level F1. Arrows indicate whether higher (↑) or lower (↓) values are better. Source Grounding is evaluated by demand coverage and GPT-4o demand–span support judgments, with GPT-4o-mini judging the same samples independently. The full audit protocol is provided in Appendix A.5.

Implementation details. Qwen3-32B (Qwen Team, 2025) constructs an OpenIE graph with passage and entity nodes connected by extracted relations, passage–entity edges, and entity-similarity edges. At query time, it runs the three OriginRAG stages. $\mathrm{Retrieve}_{\mathcal{G}}$ uses HippoRAG 2 PPR with a damping factor of 0.5, and the matched baseline uses the same graph index and retrieval settings. NV-Embed-v2 (Lee et al., 2025a) supplies dense embeddings, and GPT-4o-mini (OpenAI, 2024) serves as the QA reader. All generative calls use temperature 0. Runnable baselines use the best settings reported in their papers. All rerun methods use the same corpus, query split, and GPT-4o-mini reader. ETS-7B and Relink use the published results obtained with their native readers. Full prompts and inference settings appear in Appendices A and B.

5.2 Main Results (RQ1)

As shown in Table 1, OriginRAG achieves the highest macro-averaged retrieval and end-to-end answer scores across the three benchmarks: 89.65 R@5, 59.07 EM, and 70.78 F1. Figure 3 compares

Table 1: Main results. Average denotes the three-dataset macro-average. † marks published answer metrics with native readers. Best and second-best scores are highlighted.

	HotpotQA			2WikiMultiHopQA			MuSiQue			Average		
Method	**R@5 ↑**	**EM ↑**	**F1 ↑**	**R@5 ↑**	**EM ↑**	**F1 ↑**	**R@5 ↑**	**EM ↑**	**F1 ↑**	**R@5 ↑**	**EM ↑**	**F1 ↑**
LLM-based retrieval												
NV-Embed-v2 (dense)	93.40	59.50	73.00	75.90	55.00	60.50	68.30	35.20	46.20	79.20	49.90	59.90
Conventional RAG												
BM25	72.90	49.40	61.00	65.80	45.10	49.00	47.80	20.50	28.00	62.17	38.33	46.00
RAPTOR	90.90	59.00	72.00	71.70	49.60	54.40	65.20	31.50	41.50	75.93	46.70	55.97
LogicRAG	94.80	57.90	72.10	90.20	60.50	69.60	73.50	36.10	50.40	86.17	51.50	64.03
GraphRAG												
HippoRAG 2	95.30	58.40	72.60	89.78	60.10	69.18	73.04	35.70	49.92	86.04	51.40	63.90
HGRAG	95.50	58.70	73.90	81.00	56.50	63.00	70.10	36.80	48.10	82.20	50.67	61.67
PropRAG	96.25	58.10	73.97	92.95	61.30	69.01	73.21	35.00	49.69	87.47	51.47	64.22
G-reasoner	97.00	59.00	74.00	93.70	70.00	78.10	74.90	37.50	51.50	88.53	55.50	67.87
Evidence selection methods												
ETS-7B†	–	60.50	57.72	–	60.50	54.82	–	40.00	40.84	–	53.67	51.13
Relink†	–	55.80	70.40	–	62.80	72.20	–	30.40	41.30	–	49.67	61.30
NeocorRAG	96.00	58.40	73.67	90.90	60.80	68.50	73.10	37.10	48.10	86.67	52.10	63.42
OriginRAG	**98.15**	**62.80**	**76.73**	**94.03**	**71.20**	**79.86**	**76.78**	**43.20**	**55.76**	**89.65**	**59.07**	**70.78**

OriginRAG with leading GraphRAG methods on R@5 and F1 across the three datasets. Each axis is scaled to its observed range.

Compared with LLM-based retrieval and conventional RAG. The largest margins occur on 2WikiMultiHopQA. OriginRAG surpasses NV-Embed-v2 by 18.13 R@5, 16.20 EM, and 19.36 F1 points. The corresponding gains over RAPTOR are 22.33, 21.60, and 25.46 points. Against dependency-guided LogicRAG, the macro-average gains are 3.49 R@5, 7.57 EM, and 6.75 F1 points.

Compared with leading GraphRAG baselines. OriginRAG ranks first in all nine dataset–metric cells among GraphRAG systems and achieves the highest R@5 on every dataset. The macro-average gains over the mean of the four GraphRAG baselines are 3.59 R@5, 6.81 EM, and 6.37 F1 points. On 2WikiMultiHopQA, OriginRAG exceeds the best GraphRAG EM and F1 scores by 1.20 and 1.76 points, respectively.

Relative to HippoRAG 2, the R@5 gains are 2.85 points on HotpotQA, 4.25 on 2WikiMultiHopQA, and 3.74 on MuSiQue. The corresponding end-to-end F1 gains are 4.13, 10.68, and 5.84 points. The largest retrieval gain and answer gain therefore coincide on 2WikiMultiHopQA, where the selected and grounded two-hop evidence has the strongest effect. HotpotQA is closer to retrieval saturation: OriginRAG exceeds the strongest GraphRAG R@5 by 1.15 points while improving the strongest GraphRAG EM by 3.80 points. The disparity between retrieval and answer gains suggests that source grounding contributes to answer accuracy beyond improvements in document recall.

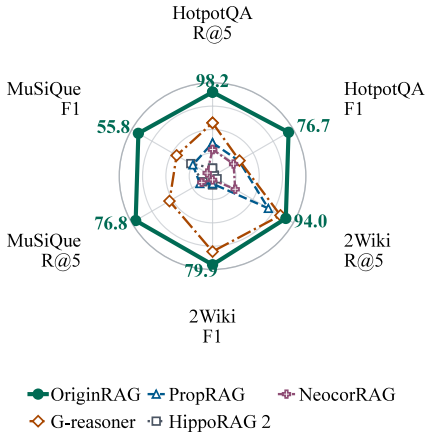


Figure 3: Relative R@5 and F1 profiles across three multi-hop QA benchmarks. Each axis is scaled to its observed range. Values mark OriginRAG results.

Compared with evidence-selection baselines. Among evidence-selection baselines under the aligned reader protocol, NeocorRAG is the strongest directly comparable system. OriginRAG raises average R@5 from 86.67 to 89.65, average EM from 52.10 to 59.07, and average F1 from 63.42 to 70.78. These gains are 2.99, 6.97, and 7.36 points, respectively. The answer gains exceed the retrieval

gain, indicating that grounding makes the selected evidence more useful to the QA reader. ETS-7B and Relink are included as published native-reader results, with macro EM/F1 scores of 53.67/51.13 and 49.67/61.30, respectively.

5.3 Ablation Study (RQ2)

Table 2: Ablation study of OriginRAG's components.

	HotpotQA			2WikiMultiHopQA			MuSiQue		
Variant	R@5 ↑	EM ↑	F1 ↑	R@5 ↑	EM ↑	F1 ↑	R@5 ↑	EM ↑	F1 ↑
Full OriginRAG	**98.15**	**62.80**	**76.73**	**94.03**	**71.20**	**79.86**	**76.78**	**43.20**	**55.76**
w/o Evidence-Demand Induction	95.65	60.80	74.73	91.53	69.20	77.86	74.28	40.70	53.76
w/o Lineage Expansion	93.15	60.30	74.23	89.03	68.70	77.36	71.78	40.20	53.26
w/o Lineage Completion	96.65	59.30	73.23	92.53	66.50	75.26	75.28	38.00	50.96

Table 2 reports the three component ablations across the benchmarks. Without Evidence-Demand Induction, the original question serves as one Evidence Demand. Without Lineage Expansion, the induced demands remain, but candidates are retrieved with the whole question and enter evidence selection without demand–passage provenance. The Lineage Completion ablation removes the completion stage, changing both the selected passage set and the grounded annotations supplied to the reader. Its retrieval and answer metrics therefore measure the combined effect of removing this stage.

Removing Evidence-Demand Induction lowers R@5 by 2.50 points and F1 by 2.00 points on each benchmark. EM falls by 2.00 points on HotpotQA and 2WikiMultiHopQA, and by 2.50 points on MuSiQue. Removing Lineage Expansion lowers R@5 by 5.00 points and F1 by 2.50 points on each benchmark, with EM losses of 2.50, 2.50, and 3.00 points. In the Lineage Completion row, R@5 falls by 1.50 points on each benchmark. EM decreases by 3.50, 4.70, and 5.20 points, and F1 by 3.50, 4.60, and 4.80 points on HotpotQA, 2WikiMultiHopQA, and MuSiQue, respectively.

5.4 Mechanism Analysis (RQ3)

OriginRAG constructs a complementary evidence set from demand–passage provenance and completes supported lineages through Source Grounding. RQ3 examines retrieval, QA, and provenance effects across the three benchmarks, with candidate-availability controls and demand–span support audits on MuSiQue.

Table 3: Attribution of evidence-set construction gains.

Method	Avg R@5 ↑	Avg EM ↑	Avg F1 ↑
$\mathrm{Retrieve}_{\mathcal{G}}$	86.04	51.40	63.90
+ SETR	86.94	52.90	64.73
+ Evidence-set construction	**88.15**	**54.60**	**66.48**
Total gain (Δ)	**+2.11**	**+3.20**	**+2.58**

Evidence-set construction. ***Does set-wise selection alone explain the retrieval gain?*** To assess the contribution of set-wise selection, we apply SETR (Lee et al., 2025b) to the candidate pool of $\mathrm{Retrieve}_{\mathcal{G}}$ as a selection-only control. Table 3 reports macro-averaged retrieval and QA scores across HotpotQA, 2WikiMultiHopQA, and MuSiQue. Evidence-set construction achieves 88.15 R@5, 54.60 EM, and 66.48 F1, exceeding the SETR control by 1.21, 1.70, and 1.75 points, respectively. Evidence-set construction introduces a new approach to evidence set selection for retrieval by coupling demand-conditioned candidate discovery with provenance-aware selection. Evidence-Demand Induction guides retrieval toward distinct information needs, while each candidate's demand associations and within-demand ranks inform the selection of complementary evidence. This connects candidate discovery with the evidence requirements that the selected set must satisfy. Table 4 examines the resulting improvements in candidate coverage and reductions in selection loss, while Table 5 isolates the contribution of correct demand–passage provenance.

Does evidence-set construction genuinely improve candidate coverage and evidence retention? Table 4 tests whether evidence-set construction offers benefits beyond combining QD and SETR on MuSiQue. QD follows Ammann et al. (2025). QD + SETR supplies merged candidates and requirements without candidate-specific demand–passage provenance. Evidence-Demand Induction extends decomposition beyond query generation by defining evidence requirements that guide retrieval, evidence set selection, and grounding. Each demand specifies what source evidence must establish, and its identity is preserved as candidate passages are retrieved and selected. This allows evidence set selection to use both passage content and demand-specific retrieval origins when assembling complementary support. Relative to QD, QD + SETR improves pool-oracle R@5 by 1.00 point and selected R@5 by 1.25 points. The composition improves coverage and retention, but its selection loss remains 10.15 points.

Pool oracle measures support available before selection. Selected reports R@5 after retaining five passages, and selection loss is their difference. Compared with QD + SETR, OriginRAG increases pool-oracle R@5 by 1.00 point and reduces selection loss by 1.43 points, achieving 76.78 selected R@5, a gain of 2.43 points. Table 5 further examines the role of provenance through removal, neutral replacement, and shuffling controls across the three benchmarks.

Table 4: Evidence coverage and retention.

Pipeline	Pool oracle R@5 ↑	Selected R@5 ↑	Selection loss ↓
QD	83.50	73.10	10.40
SETR	83.60	73.30	10.30
QD + SETR	84.50	74.35	10.15
OriginRAG	**85.50**	**76.78**	**8.72**

Does evidence selection depend on correct demand–passage provenance? Table 5 reports macro-averaged R@5 across the three benchmarks. For each query, all conditions use the same candidate pool, selector, and five-passage budget, varying only the provenance shown for each candidate. Removing provenance lowers average R@5 from 89.65 to 86.15. Equal-length neutral text gives 86.05, only 0.10 points lower, so extra prompt content does not explain the gain. Shuffling valid provenance across passages preserves its volume but breaks the candidate-specific correspondence. Average R@5 falls to 80.45, which is 9.20 points below correct provenance and 5.70 points below removing it. This ordering is diagnostic. Missing provenance removes demand-coverage information, while incorrect provenance actively directs its five positions toward the wrong passages. Correct provenance lets the selector allocate those positions across distinct Evidence Demands and retain complementary support. Read with Table 4, these controls attribute the selection-side benefit of provenance lineage to preserving candidate-specific demand–passage correspondence between Demand-Conditioned Lineage Expansion and evidence set selection.

Table 5: Effect of demand–passage provenance on evidence selection.

Demand–passage provenance	Avg R@5 ↑	Δ
Correct (OriginRAG)	**89.65**	–
Removed	86.15	-3.50
Equal-length neutral replacement	86.05	-3.60
Shuffled across passages	80.45	-9.20

Table 6: Lineage completion and demand–span support.

Provenance-lineage completion		*Demand–span support*	
Measure	Result	Measure	Result
Grounded Evidence Demands	1,865 / 2,182	Fully supporting spans	420 / 500
Queries with all demands grounded	734 / 1,000	Full or partial support	470 / 500
Queries with any demand grounded	976 / 1,000	Wrong-demand links	10 / 500

Source Grounding. ***How often does Source Grounding complete a provenance lineage?*** Table 6 measures whether the lineage established for each Evidence Demand reaches a verbatim span in the selected source. Source Grounding completes 1,865 of 2,182 induced lineages. All lineages are completed for 734 of 1,000 queries, and 976 queries contain at least one completed lineage. Of the 266 queries without full completion, 242 contain both completed and open lineages, while 24 contain no completed lineage. Open lineages are thus concentrated within partially completed queries. Demand-level analysis identifies requirements still ungrounded after evidence-set construction.

Do the grounded spans support their linked Evidence Demands? Table 6 evaluates whether each grounded endpoint establishes its linked demand. Of the 500 audited spans, 420 provide full support

and 50 provide partial support. The partial cases reach relevant source evidence but do not establish the complete fact specified by the demand. Among the remaining 30 records, 20 do not establish the required fact, while 10 attach a span to the wrong demand. These errors distinguish insufficient source evidence from an incorrect demand–span connection. The counts expose span adequacy and correspondence as separate factors. GPT-4o-mini matches GPT-4o on 481 of 500 binary support decisions (96.2%), indicating that the distinction between supporting and insufficient evidence is stable across the two judges. See Appendix A.5 for the audit protocol.

Does Source Grounding improve answer generation? Table 7 reports a fixed-passage reader comparison, with QA scores macro-averaged across HotpotQA, 2WikiMultiHopQA, and MuSiQue. For each query, both conditions use the same selected passages and QA reader settings, differing only in whether grounded spans are included. Adding grounded spans raises EM from 54.60 to 59.07 and F1 from 66.48 to 70.78, yielding gains of 4.47 EM and 4.30 F1 points. Means and gains are computed before rounding. These results show that demand-linked grounded spans improve answer generation from the same retrieved evidence.

Table 7: Source Grounding for QA.

Reader input	Avg EM ↑	Avg F1 ↑
Selected passages	54.60	66.48
Selected passages with grounded spans	**59.07**	**70.78**
Gain	**+4.47**	**+4.30**

The mechanism analysis follows provenance lineage from evidence-set construction to answer generation. Tables 3–5 examine the role of demand–passage provenance in selecting evidence. Table 7 isolates the reader-side effect of grounded spans while holding the selected passages fixed. Together, these analyses examine evidence selection and the use of selected evidence for QA.

5.5 Computational Cost (RQ4)

Table 8 separates one-time graph construction from query-time model use. OriginRAG and HippoRAG 2 share the same graph index, whose construction consumes 13.49M tokens. HippoRAG 2 follows one whole-query retrieval path. OriginRAG additionally induces demands, retrieves for each demand, selects from a provenance-augmented pool, and extracts grounded spans for the reader. These stages preserve demand identity from candidate discovery through source grounding and increase query-time token consumption from 0.63M to 1.54M tokens per 1,000 queries, or 910 additional tokens per query. The total for one graph construction and 1,000 queries rises from 14.12M to 15.03M. Because graph construction is shared, the total difference comes from these query-time lineage operations.

Table 8: Average model-token consumption across the three benchmarks.

Method	Offline (M)	Query-time (M / 1,000 queries)	Total (M)
OriginRAG	13.49	1.54	15.03
HippoRAG 2	13.49	0.63	14.12
PropRAG	18.84	0.00	18.84
NeocorRAG	12.84	6.94	19.78

This 2.44-fold increase in query-time token consumption accompanies gains of 3.61 R@5, 7.67 EM, and 6.88 F1 over HippoRAG 2. The Lineage Expansion ablation in Table 2 lowers R@5 by 5.00 points on each benchmark. The fixed-passage reader comparison in Table 7 shows macro-average gains of 4.47 EM and 4.30 F1 from adding grounded spans. These gains locate the return from the additional computation in both evidence construction and answer generation. PropRAG uses 18.84M tokens offline and none online, while NeocorRAG uses 6.94M online.

6 Conclusion

In this work, we propose OriginRAG, a novel framework that harnesses GraphRAG by maintaining a set of demand-specific provenance lineages as its retrieval state. This design directly addresses the missing connection between evidence discovery and source grounding in multi-hop reasoning. OriginRAG induces Evidence Demands and expands each through graph retrieval while preserving demand–passage provenance. It then uses this provenance to construct a complementary evidence set and ground supported demands in source text. Experiments on three multi-hop QA benchmarks show improved retrieval and answer quality over leading GraphRAG baselines and demonstrate the value of provenance lineage for source-grounded multi-hop reasoning.

Ethics Statement

This work uses public multi-hop QA benchmarks and collects no private user data. The main ethical risks arise from inaccuracies or biases in the underlying corpora and language models, which may lead to unsupported or misleading answers. Tracing evidence origins and grounding demands in verbatim spans improve auditability without guaranteeing factual correctness. Sensitive applications therefore require expert review of the retrieved evidence and generated answers. We report model-token consumption to disclose the computational cost of demand-specific retrieval, evidence selection, and source grounding.

Reproducibility Statement

To support reproducibility, we provide detailed descriptions of the method and experiments in the paper and appendices. All datasets used in this study are publicly available. Section 4 describes the retrieval pipeline and core prompts, while Section 5.1 specifies the datasets, evaluation protocols, model choices, and retrieval settings. Appendices A and B provide additional implementation details. Our code is provided in the supplementary material and will be open-sourced upon acceptance.

Use of LLM

During the preparation of this manuscript, a large language model assisted with language polishing. The authors retain full responsibility for the final manuscript and for verifying all statements, results, and conclusions.

References

Paul J. L. Ammann, Jonas Golde, and Alan Akbik. Question decomposition for retrieval-augmented generation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 4: Student Research Workshop)*, pp. 497–507, 2025. doi: 10.18653/v1/2025.acl-srw.32. URL https://aclanthology.org/2025.acl-srw.32/.

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. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(36):30270–30278, 2026. doi: 10.1609/aaai.v40i36.40278. URL https://ojs.aaai.org/index.php/AAAI/article/view/40278.

Yingwei Cui, Jennifer Widom, and Janet L. Wiener. Tracing the lineage of view data in a warehousing environment. *ACM Transactions on Database Systems*, 25(2):179–227, 2000. doi: 10.1145/357775.357777. URL https://doi.org/10.1145/357775.357777.

Zirui Guo, Lianghao Xia, Yanhua Yu, Tu Ao, and Chao Huang. LightRAG: Simple and fast retrieval-augmented generation. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pp. 10746–10761, 2025. doi: 10.18653/v1/2025.findings-emnlp.568. URL https://aclanthology.org/2025.findings-emnlp.568/.

Bernal Jiménez Gutiérrez, Yiheng Shu, Yu Gu, Michihiro Yasunaga, and Yu Su. HippoRAG: Neurobiologically inspired long-term memory for large language models. In *Advances in Neural Information Processing Systems*, volume 37, pp. 59532–59569. Curran Associates, Inc., 2024. doi: 10.52202/079017-1902. URL https://proceedings.neurips.cc/paper_files/paper/2024/hash/6ddc001d07ca4f319af96a3024f6dbd1-Abstract-Conference.html.

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 *Proceedings of the 42nd International Conference on Machine Learning*, volume 267 of *Proceedings of Machine Learning Research*, pp. 21497–21515, 2025. URL https://proceedings.mlr.press/v267/gutierrez25a.html.

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. doi: 10.18653/v1/2020.coling-main.580. URL https://aclanthology.org/2020.coling-main.580/.

Manzong Huang, Chenyang Bu, Yi He, Xingrui Zhuo, and Xindong Wu. Relink: Constructing query-driven evidence graph on-the-fly for GraphRAG. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(37):31202–31210, 2026. doi: 10.1609/aaai.v40i37.40382. URL https://doi.org/10.1609/aaai.v40i37.40382.

Kai Kretzschmar and Fiona M. Watt. Lineage tracing. *Cell*, 148(1–2):33–45, 2012. doi: 10.1016/j.cell.2012.01.002. URL https://doi.org/10.1016/j.cell.2012.01.002.

Chankyu Lee, Rajarshi Roy, Mengyao Xu, Jonathan Raiman, Mohammad Shoeybi, Bryan Catanzaro, and Wei Ping. NV-embed: Improved techniques for training LLMs as generalist embedding models. In *The Thirteenth International Conference on Learning Representations*, 2025a. URL https://openreview.net/forum?id=lgsyLSsDRe.

Dahyun Lee, Yongrae Jo, Haeju Park, and Moontae Lee. Shifting from ranking to set selection for retrieval augmented generation. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 17606–17619, 2025b. doi: 10.18653/v1/2025.acl-long.861. URL https://aclanthology.org/2025.acl-long.861/.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems*, volume 33, pp. 9459–9474. Curran Associates, Inc., 2020. URL https://proceedings.neurips.cc/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html.

Jiaoyang Li, Junhao Ruan, Shengwei Tang, Saihan Chen, Kaiyan Chang, Yuan Ge, Tong Xiao, and Jingbo Zhu. SUBQRAG: Sub-question driven dynamic graph RAG. *arXiv preprint arXiv:2510.07718*, 2025a. doi: 10.48550/arXiv.2510.07718. URL https://arxiv.org/abs/2510.07718.

Minghan Li, Junjie Zou, Xinxuan Lv, Chao Zhang, and Guodong Zhou. S2G-RAG: Structured sufficiency and gap judging for iterative retrieval-augmented QA. In *Proceedings of the 64th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 25846–25862. Association for Computational Linguistics, 2026. doi: 10.18653/v1/2026.acl-long.1185. URL https://aclanthology.org/2026.acl-long.1185/.

Mufei Li, Siqi Miao, and Pan Li. Simple is effective: The roles of graphs and large language models in knowledge-graph-based retrieval-augmented generation. In *The Thirteenth International Conference on Learning Representations*, 2025b. URL https://openreview.net/forum?id=JvkuZZ04O7.

Linhao Luo, Zicheng Zhao, Junnan Liu, Zhangchi Qiu, Junnan Dong, Serge Panev, Chen Gong, Thuy-Trang Vu, Gholamreza Haffari, Dinh Phung, Alan Wee-Chung Liew, and Shirui Pan. G-reasoner: Foundation models for unified reasoning over graph-structured knowledge. In *The Fourteenth International Conference on Learning Representations*, 2026. URL https://arxiv.org/abs/2509.24276.

OpenAI. GPT-4o mini: Advancing cost-efficient intelligence. OpenAI Research, 2024. URL https://openai.com/index/gpt-4o-mini-advancing-cost-efficient-intelligence/.

Shiyao Peng, Qianhe Zheng, Zhuodi Hao, Zichen Tang, Rongjin Li, Qing Huang, Jiayu Huang, Jiacheng Liu, Yifan Zhu, and Haihong E. NeocorRAG: Less irrelevant information, more explicit evidence, and more effective recall via evidence chains. In *Proceedings of the ACM Web Conference 2026*, pp. 1899–1910. ACM, 2026. doi: 10.1145/3774904.3792093. URL https://doi.org/10.1145/3774904.3792093.

Qwen Team. Qwen3 technical report. *arXiv preprint arXiv:2505.09388*, 2025. doi: 10.48550/arXiv.2505.09388. URL https://arxiv.org/abs/2505.09388.

Stephen Robertson and Hugo Zaragoza. The probabilistic relevance framework: BM25 and beyond. *Foundations and Trends in Information Retrieval*, 3(4):333–389, 2009. doi: 10.1561/1500000019. URL https://doi.org/10.1561/1500000019.

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.

Hao Sun, Hengyi Cai, Yuchen Li, Xuanbo Fan, Xiaochi Wei, Shuaiqiang Wang, Yan Zhang, and Dawei Yin. Enhancing retrieval-augmented generation via evidence tree search. In *Proceedings of the 63rd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 24116–24127. Association for Computational Linguistics, 2025. doi: 10.18653/v1/2025.acl-long.1175. URL https://aclanthology.org/2025.acl-long.1175/.

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. doi: 10.1162/tacl_a_00475. URL https://aclanthology.org/2022.tacl-1.31/.

Changjian Wang, Weihong Deng, Weili Guan, Quan Lu, and Ning Jiang. Cross-granularity hypergraph retrieval-augmented generation for multi-hop question answering. *Proceedings of the AAAI Conference on Artificial Intelligence*, 40(39):33368–33376, 2026a. doi: 10.1609/aaai.v40i39.40623. URL https://doi.org/10.1609/aaai.v40i39.40623.

Jingjin Wang and Jiawei Han. PropRAG: Guiding retrieval with beam search over proposition paths. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pp. 6212–6227, 2025. doi: 10.18653/v1/2025.emnlp-main.317. URL https://aclanthology.org/2025.emnlp-main.317/.

Shuting Wang, Xin Yu, Mang Wang, Weipeng Chen, Yutao Zhu, and Zhicheng Dou. RichRAG: Crafting rich responses for multi-faceted queries in retrieval-augmented generation. In *Proceedings of the 31st International Conference on Computational Linguistics*, pp. 11317–11333. Association for Computational Linguistics, 2025. URL https://aclanthology.org/2025.coling-main.750/.

Yubo Wang, Haoyang Li, Fei Teng, and Lei Chen. AGRAG: Advanced graph-based retrieval-augmented generation for LLMs. In *2026 IEEE 42nd International Conference on Data Engineering (ICDE)*, pp. 1–15. IEEE, 2026b. doi: 10.1109/ICDE65706.2026.00110. URL https://doi.org/10.1109/ICDE65706.2026.00110.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W. 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. doi: 10.18653/v1/D18-1259. URL https://aclanthology.org/D18-1259/.

Jungmin Yun and Youngbin Kim. Query, decompose, compress: Structured query expansion for efficient multi-hop retrieval. In *Proceedings of the 34th ACM International Conference on Information and Knowledge Management*, pp. 5459–5463, 2025. doi: 10.1145/3746252.3760968.

A PROMPT TEMPLATES

The following instructions describe each inference stage. Repeated input wrappers are omitted.

A.1 EVIDENCE-DEMAND INDUCTION

The induction call receives only the question.

Evidence-Demand Induction

You decompose multi-hop QA questions into Evidence Demands. An Evidence Demand describes what a retrieved evidence unit must establish for the answer to be derivable. Do not answer the question. Do not use dataset names. Do not mention gold passages. Use only the question text. Do not introduce named entities, titles, dates, numbers, or answer values that are absent from the question. If an entity must be discovered by retrieval, describe it as a variable such as "the bridge entity", "that person", "that organization", or "the answer entity".

Demands are induced jointly. Each retrieval uses the question and one demand independently.

A.2 LINEAGE COMPLETION: EVIDENCE SET SELECTION

The selector receives the question, induced demands, and candidate excerpts augmented with the provenance representation $b_q(d)$.

Evidence Set Selection

Select exactly k passages that jointly cover the evidence demands. Prefer complementary evidence and avoid redundant passages that repeat a covered demand. Preserve useful direct evidence, and prioritize answer-bearing passages only when they are connected to the demands. Return the selected candidate IDs.

A.3 LINEAGE COMPLETION: SOURCE GROUNDING

Grounding processes the selected passages and demands in one joint call.

Source Grounding

For each supported demand, identify its selected source passage and copy the supporting span verbatim. Extract relations and variable bindings only from that span, preserving the demand identifier. Use no outside knowledge and return no annotation for an unsupported demand. Keep distinct source-local annotations separate.

A.4 QA READER

QA Reader

Answer with the shortest exact phrase supported by the selected passages. Use grounded annotations only as an index into those passages. Do not explain. Return exactly one line: `Answer: <short phrase>`.

A.5 DEMAND–SPAN SEMANTIC AUDIT

The audit samples one retained annotation from each of 500 distinct MuSiQue queries using random seed 1729. GPT-4o-2024-08-06 is the primary judge. GPT-4o-mini-2024-07-18 independently judges the same sample.

Each judge evaluates whether the exact source span establishes its linked demand. The question and source title may disambiguate the subject but may not supply missing relations or values. Full support requires the complete demand to be established. Partial support requires a necessary part. Incorrect subject or input bindings are judged separately. Gold answers and benchmark support annotations are excluded.

GPT-4o labels 420 spans as fully supporting and 50 as partially supporting. Of the remaining 30 spans, 20 are unsupported and 10 are linked to the wrong demand. For binary agreement, full and partial support are grouped together. GPT-4o labels 470 records as support and 30 as insufficient, while GPT-4o-mini labels 469 as support and 31 as insufficient. The judges agree on 481 of 500 decisions (96.2%), with Cohen's $\kappa = 0.67$.

B Inference Configuration

OriginRAG performs inference without parameter updates. All rerun baselines use the same corpus, split, GPT-4o-mini reader, and author-reported settings without evaluation-query retuning. ETS-7B and Relink use published native-reader results.

Token counts in Table 8 exclude the final QA reader.

For SETR, we use its reported settings and evaluate the first five selected passages. QD + SETR supplies merged QD candidates and requirements without candidate-level demand–passage provenance.

We set $m_{\max} = 5$. Successful induction returns two to five demands.

Table 9: Corpus-graph statistics for the three benchmarks.

Dataset	Passages	Entities	Triples	Edges
HotpotQA	9,811	82,440	122,749	1,385,812
2Wiki	6,119	46,927	68,238	897,016
MuSiQue	21,100	143,334	221,343	2,967,364

Table 10: Configuration used in the reported experiments.

Graph retrieval	**Setting**
Offline graph construction	Qwen3-32B OpenIE
Graph nodes	Passage and entity nodes
Graph edges	OpenIE relation, passage–entity, and entity-similarity edges
Graph direction	Undirected
Entity-similarity threshold	0.8
Graph propagation	HippoRAG 2 PPR, damping 0.5
Fact seeds per demand	Up to 5
Passage seeds per demand	5, weight 0.05
Dense embeddings	NV-Embed-v2
Graph output per demand	50 passages
Candidates per demand	12
Merged candidate limit	24
Candidate merging	Rank-depth traversal with first-occurrence deduplication

Online inference	**Setting**
Evidence-demand induction	Qwen3-32B, temperature 0
Graph-retrieval inference	Qwen3-32B, temperature 0
Evidence set selector	Qwen3-32B, temperature 0
Selected context size	$k = 5$
Candidate excerpt limit	760 characters
Grounding text limit	900 characters
Source grounding	Qwen3-32B, temperature 0
QA reader	GPT-4o-mini, temperature 0