Rethinking Retrieval in RAG: Recall, Context Length, and Efficient Multi-Hop Reasoning

Anonymous ACL submission

Abstract

Retrieval-Augmented Generation (RAG) is widely used to ground large language models in external evidence, yet its effectiveness in multi-hop reasoning is often limited by retrieval design and evaluation practices. We conduct a systematic analysis of RAG, examining chunk size, document transformation, and context length under a fixed token budget. Contrary to the common assumption that more sophisticated transformation- or LLM-dependent retrieval pipelines necessarily improve multi-hop RAG, our results show that widely used document transformations often discard, distort, or dilute critical evidence, thereby degrading reasoning performance under realistic context budgets. These results suggest that, under realistic context and cost constraints, the primary bottleneck of RAG is not the lack of increasingly sophisticated retrieval or transformation modules, but the failure to preserve sufficient supporting evidence within the final context. Guided by these insights, we formulate a set of general and practical design principles for practical RAG systems and propose a recall-oriented RAG framework with fine-grained chunking, an LLM-free Query Expansion, and contextual reranking. Experiments on three domain-specific and four multi-hop benchmarks demonstrate that our method outperforms competitive baselines while significantly reducing latency and token cost. Our code is available at: https://anonymous.4open.science/r/RAG-FOUNDATION-D1A7

1 Introduction

Despite the progress of large language models (LLMs), hallucination (Huang et al., 2025; Zhang et al., 2025), outdated knowledge (Tonmoy et al., 2024; Xu et al., 2024), and weak grounding (Liu, 2024) remain persistent limitations. Retrieval-Augmented Generation (RAG) addresses these issues by conditioning generation on external evidence, and has become central to domain-specific and multi-hop QA (Ru et al., 2024; Yang et al., 2024). However, as RAG moves beyond simple open-domain settings, retrieval design choices such as chunk granularity, context length, and budget allocation become increasingly decisive for final answer quality.

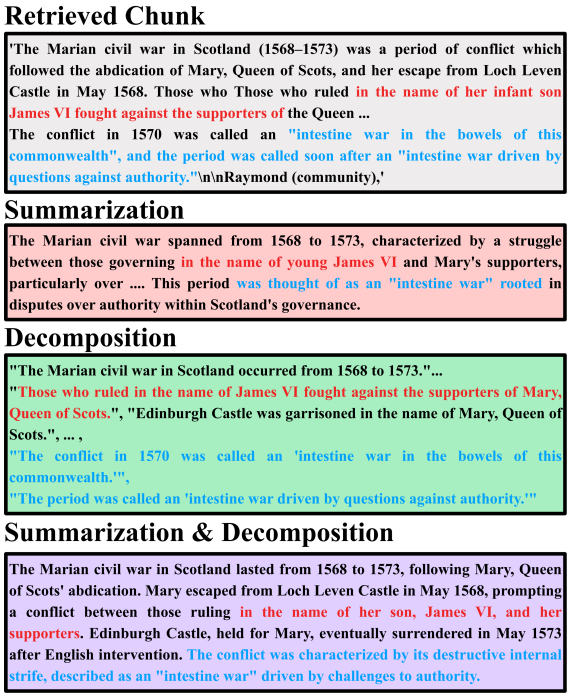


Figure 1: An illustration of document transformation effects in RAG. As raw retrieved chunks undergo summarization, decomposition, and their combination, important details may be omitted, altered, or rephrased inconsistently, introducing noise and missing information—issues that are particularly harmful for multi-hop reasoning, where precise factual dependencies across evidence are critical.

To address complex reasoning requirements, many existing RAG approaches rely on aggressive document manipulation or expanded contexts, including summarization (Sarthi et al.; Achkar et al., 2025; Hong et al., 2025), decomposition (Guo et al., 2025; Gutiérrez et al.; Luo et al., 2025), paraphrasing (Ji et al., 2024; Xian et al., 2025), or their

combinations (Edge et al., 2024), often followed by concatenating transformed outputs with original documents. These pipelines frequently employ extremely long contexts—sometimes approaching 20,000 tokens—or perform multi-step, iterative generation to simulate reasoning processes (Press et al., 2023; Shao et al., 2023). However, such designs introduce several limitations. As illustrated in Figure 1, repeated document transformations such as summarization and decomposition can omit, distort, or inconsistently rephrase factual details, introducing noise and missing information that are especially harmful for multi-hop reasoning. Prior work (Laitenberger et al., 2025) has shown that contexts closer to original documents often yield better performance than heavily transformed representations (Lossy Conversion), while studies on the Lost in the Middle (Liu et al., 2024) phenomenon demonstrate that excessively long contexts (e.g., beyond 5,000 tokens) can cause models to overlook relevant information. Moreover, iterative generation pipelines incur substantial latency and token costs, which become a critical bottleneck in practical RAG deployments. Despite these drawbacks, most evaluations still rely on top-k retrieved documents as the primary comparison unit, obscuring the sensitivity of performance to chunk size, context construction, and cost-efficiency trade-offs.

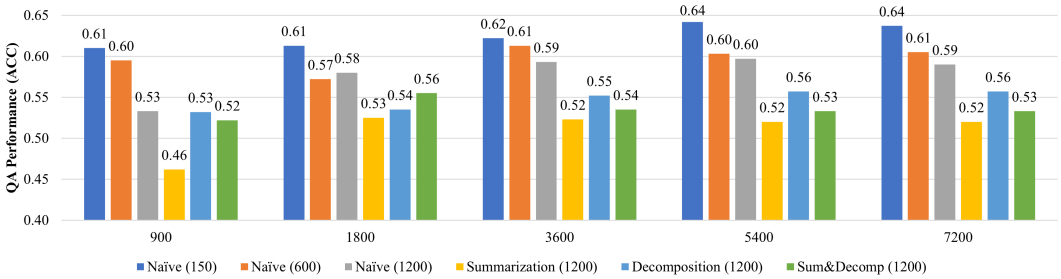
We conduct a controlled empirical analysis of multi-hop RAG to examine Lost in the Middle and Lossy Conversion, comparing naïve retrieval with varying chunk sizes against summarization- and decomposition-based pipelines under a fixed token-level context budget. Our results reveal three consistent findings. First, top-k-based evaluation is highly sensitive to chunk size, leading to unstable and sometimes misleading performance comparisons. Second, document transformations—including summarization, rewriting, compression, and paraphrasing—introduce noise and information loss that degrade answer accuracy, particularly in multi-hop reasoning scenarios. Third, retrieval recall emerges as the dominant factor determining final performance, outweighing the benefits of additional reasoning steps or complex generation pipelines. These findings suggest that many existing RAG approaches unintentionally sacrifice recall and factual completeness, while incurring significant computational overhead.

Guided by these empirical observations, we propose a new RAG design principle that aims to maximize retrieval recall under a fixed context length while minimizing noise and avoiding costly iterative reasoning. Our approach adopts a fine-grained chunking strategy that enables the retrieval of a larger number of relevant evidence units within the same token budget, thereby improving recall without increasing context length. To efficiently explore multi-hop reasoning paths, we formulate our Query Expansion as an LLM-free pseudo-relevance feedback (PRF)-style process. Instead of generating intermediate sub-questions with an LLM, we treat initially retrieved fine-grained chunks as pseudo-relevant evidence and use them to expand the original query for broader evidence coverage. Finally, we apply contextual reranking to reconstruct a coherent context by reconnecting consecutive chunks in their original document order, preserving retrieval ranking while remaining close to the original documents. Together, these components form an efficient multi-hop RAG pipeline that reduces latency and token cost while retaining high factual fidelity.

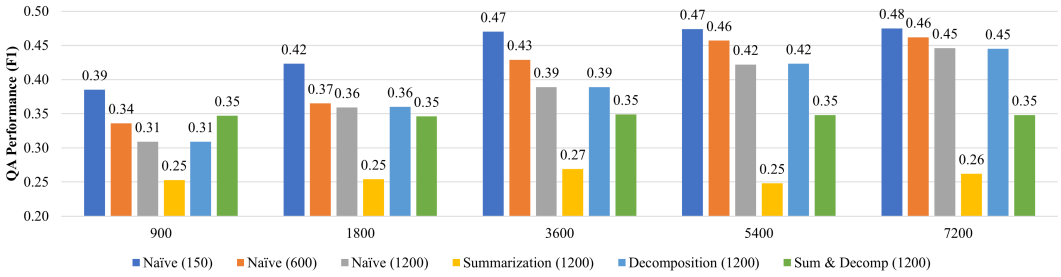
We evaluate the proposed method on three domain-specific datasets and four multi-hop reasoning benchmarks. Across all settings, our approach achieves consistently strong performance. Compared to baselines specifically designed for multi-hop reasoning, our method demonstrates superior efficiency, reducing both latency and token consumption while maintaining or improving answer accuracy.

In summary, this paper makes the following contributions:

- We systematically analyze RAG performance under fixed context budgets, showing how chunk size affects retrieval recall and answer accuracy.
- We demonstrate that common document transformation strategies can harm RAG by losing or distorting supporting evidence.
- We establish practical design principles for broadly applicable RAG systems: compare by context budget, maximize evidence recall, and minimize unnecessary transformations.
- We instantiate these principles with a lightweight LLM-free framework that achieves state-of-the-art performance across all benchmarks while reducing latency and token cost.



(a) Performance on ClapNQ dataset



(b) Performance on MuSiQue dataset

Figure 2: Effect of chunk size, context length, and document conversion on RAG performance. Under fixed context length budgets, smaller chunks yield higher QA accuracy by improving evidence coverage, while summarization and decomposition consistently degrade performance compared to naïve retrieval, especially on multi-hop reasoning (MuSiQue). Numbers in parentheses denote the chunk size measured in tokens.

Context Length	900	1800	3600	5400	7200	Correlation w/ performance
Recall						
Naïve (150)	0.461	0.593	0.678	0.719	0.749	**0.875**
Naïve (600)	0.257	0.365	0.541	0.613	0.659	**0.609**
Naïve (1200)	0.128	0.235	0.304	0.396	0.432	**0.858**
Precision						
Naïve (150)	0.363	0.243	0.142	0.101	0.079	-0.887
Naïve (600)	0.343	0.329	0.247	0.189	0.153	-0.640
Naïve (1200)	0.320	0.287	0.248	0.195	0.177	-0.792
F1						
Naïve (150)	0.398	0.338	0.230	0.174	0.141	-0.936
Naïve (600)	0.290	0.341	0.334	0.285	0.245	-0.397
Naïve (1200)	0.181	0.255	0.269	0.257	0.248	0.964

Table 1: Retrieval metrics and their correlation with QA performance on ClapNQ. Numbers in parentheses denote chunk size (tokens). Recall correlates positively with accuracy, while Precision and F1 are less reliable under longer contexts.

Context Length	900	1800	3600	5400	7200	Correlation w/ performance
Recall						
Naïve (150)	0.294	0.400	0.510	0.576	0.616	**0.970**
Naïve (600)	0.250	0.331	0.483	0.563	0.609	**0.995**
Naïve (1200)	0.123	0.233	0.315	0.421	0.457	**0.996**
Precision						
Naïve (150)	0.377	0.264	0.171	0.131	0.106	-0.985
Naïve (600)	0.611	0.550	0.413	0.327	0.269	-0.987
Naïve (1200)	0.508	0.484	0.443	0.364	0.332	-0.962
F1						
Naïve (150)	0.321	0.310	0.252	0.209	0.178	-0.891
Naïve (600)	0.345	0.401	0.432	0.402	0.118	-0.347
Naïve (1200)	0.193	0.305	0.355	0.377	0.372	0.938

Table 2: Retrieval metrics and their correlation with QA performance on MuSiQue. Numbers in parentheses denote chunk size (tokens). Recall strongly correlates with accuracy for multi-hop questions, while Precision and F1 are unreliable.

2 Pilot Study & Analysis

In this pilot study, we systematically analyze the effects of chunk size, context length, and document conversion strategies on retrieval and generation performance in RAG. Most existing RAG approaches retrieve the top-k textual units—such as passages, chunks, triplets, or atomic facts—and concatenate them into the context for answer generation. However, the actual amount of information available to the language model is determined by the total context length rather than the number of retrieved units, leading to substantial variation in effective information content depending on chunk granularity. To control for this factor, we split documents into chunks of 150, 600, and 1200 tokens, and evaluate retrieval performance (Recall, Precision, and F1) as well as downstream answer accuracy under fixed context length budgets of 900, 1800, 3600, 5400, and 7200 tokens. In addition, we examine the impact of document conversion strategies—summarization, decomposition, and their combination—applied to retrieved chunks, and compare them against a naïve baseline without modification. Experiments are conducted on ClapNQ (Rosenthal et al., 2025), representing open-domain long-answer question answering, and

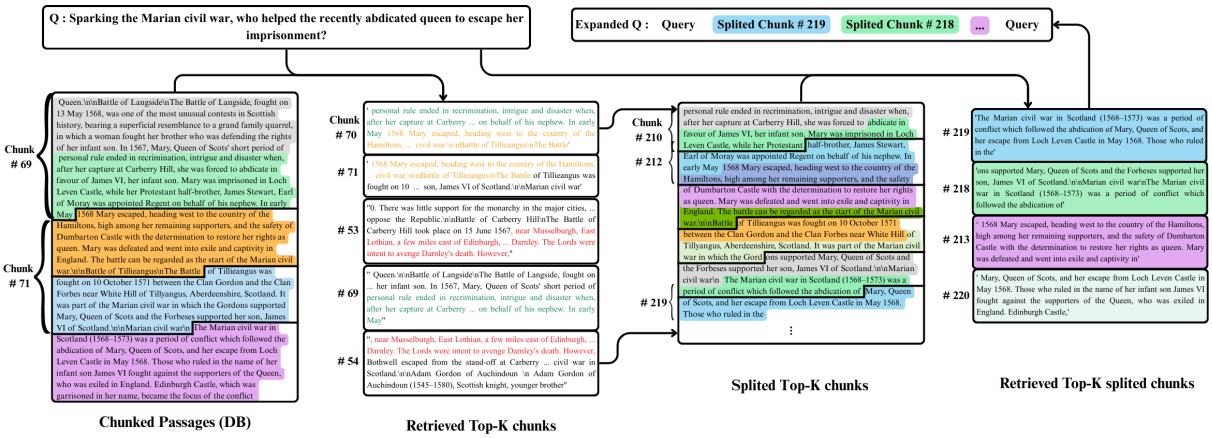


Figure 3: Illustration of LLM-free PRF-style Query Expansion. Given a multi-hop question, the original query is expanded without LLM generation, enabling retrieval of multiple relevant fine-grained chunks across documents. Retrieved chunks are split into smaller units, where background-colored regions indicate overlapping spans originating from the same source text.

MuSiQue (Trivedi et al., 2022), representing multi-hop reasoning with short answers. For summarization, we adopt the prompt used in the RAPTOR (Sarthi et al.), while the decomposition strategy is implemented using the atomic fact extraction prompt detailed in Appendix A.1.

2.1 Finding 1: chunk size and context length

Results on both ClapNQ (Figure 2a) and MuSiQue (Figure 2b) demonstrate that top-k-based retrieval performance is highly sensitive to chunk size. When evaluated under the same context length, smaller chunks consistently yield better performance by covering more relevant evidence within a fixed token budget. However, under the same top-k setting, a chunk size of 1200 tokens often matches or outperforms smaller chunk sizes across both datasets. These results indicate that top-k retrieval is insufficient for fair comparison, as it fails to control for the actual amount of information used during generation.

2.2 Finding 2: Lossy Document Conversion

Across both datasets (Figure 2), document conversion methods consistently underperform the naïve baseline, even when chunk size and context length are held constant. This indicates that summarization and decomposition introduce information loss or noise that harms answer generation. While these methods are intended to improve efficiency, they often fail to preserve fine-grained evidence required for accurate reasoning. The performance gap is particularly large on MuSiQue, where multi-hop questions rely on supporting evidence distributed across passages. These findings suggest that document transformation, as commonly used in prior work, can negatively impact RAG performance rather than improve it.

2.3 Finding 3: Recall as the Core Factor

Our correlation analysis shows that Recall is the primary factor driving generation performance in RAG. On ClapNQ (Table 1) and MuSiQue (Table 2), Recall exhibits strong positive correlations with answer accuracy, whereas Precision decreases as context length grows and F1 often shows a negative correlation. This suggests that modern LLMs can tolerate noisy contexts as long as the correct evidence is included. In other words, irrelevant chunks are less harmful than the absence or degradation of necessary evidence. This perspective also explains why document transformation methods underperform naïve retrieval: although they aim to produce cleaner or more structured contexts, summarization and decomposition can weaken effective recall by omitting, fragmenting, or distorting supporting evidence. However, beyond roughly 5400 tokens, performance saturates despite continued gains in Recall, highlighting the inefficiency of longer contexts. This behavior provides clear evidence of the Lost in the Middle effect and shows that simply increasing context length yields diminishing returns.

Our findings do not imply that document transformation is universally harmful. Rather, they show that in representative, broadly applicable RAG pipelines, commonly used transformations such as summarization, decomposition, and paraphrasing often reduce answer accuracy under fixed context budgets by weakening the preservation of supporting evidence.

Dataset	#Passages	Avg. Passage Length (tokens)	#Queries	Reasoning Type	Answer Type	Avg. Answer Length (tokens)	Source Domain
ClapNQ (Rosenthal et al., 2025)	600	223.6	600	Single-hop	Open-ended	32.0	Wikipedia
FiQA (Maia et al., 2018)	57,638	169.4	500	Single-hop	Open-ended	162.8	Finance
NovelQA (Wang et al., 2024)	19	374,592.4	300	Single-hop	Open-ended	42.2	Fiction
HotpotQA (Yang et al., 2018)	9,811	126.4	1,000	Multi-hop	Short Answer	4.0	Wikipedia
MultihopRAG (Tang and Yang)	609	2,289.80	1,000	Multi-hop	Short Answer	2.1	Wikipedia
MuSiQue (Trivedi et al., 2022)	11,656	109.9	1,000	Multi-hop	Short Answer	4.4	Wikipedia
2WikiMultiHopQA (Ho et al., 2020)	6,119	104.6	1,000	Multi-hop	Short Answer	4.1	Wikipedia

Table 3: Dataset statistics for multi-domain and multi-hop QA benchmarks used in our experiments.

3 Method

Based on the empirical analysis in the previous sections, we identify three core principles for constructing an effective and fair RAG framework: (i) the final context length should be bounded within approximately 5,400 tokens, (ii) retrieval should maximize recall under this fixed budget, and (iii) transforming retrieved documents should be avoided to prevent information loss and noise amplification. Guided by these principles, we propose a retrieval framework that preserves original documents while maximizing evidence coverage within a constrained context length, and further introduce a pseudo-relevance feedback (PRF)-style formulation of query expansion: rather than relying on LLM-generated intermediate queries, we use evidence retrieved in the first pass as pseudo-relevant feedback to improve retrieval recall in the second pass. This design increases evidence coverage and eliminates the computational and monetary overhead of iterative LLM-based generation.

To efficiently construct the final context from a large document collection, we split all documents into chunks of 150 tokens with 50% overlap, following the empirical results in Section 2. Although each chunk inevitably loses global document context, smaller chunks enable more fine-grained and information-dense retrieval with respect to the query, and overlapping boundaries reduce the risk of missing critical evidence located near chunk edges by allowing adjacent contextual spans to be retrieved together.

3.1 LLM-free PRF-style Query Expansion

Existing multi-hop retrieval and iterative generation methods typically rely on repeated LLM calls to generate intermediate queries, incurring substantial latency and token cost. To address this limitation, we propose an LLM-free PRF-style Query Expansion strategy. In pseudo-relevance feedback, top-ranked results from an initial retrieval pass are treated as pseudo-relevant evidence and used to reformulate or enrich the original query. We adapt this idea to RAG by using initially retrieved fine-grained chunks as feedback signals, rather than relying on LLM-generated sub-questions or transformed document representations. Specifically, we first retrieve top-k documents using the original query, split these documents into smaller 50-token chunks, and select the most relevant ones based on similarity to the query (Figure 3). These selected chunks are then concatenated with the original query to form an expanded query, which increases the likelihood of retrieving downstream evidence in multi-hop reasoning scenarios, such as linking a descriptive attribute to a specific entity and subsequently retrieving its associated facts, while maintaining a single-pass LLM inference.

3.2 Contextual Reranking

While reranking is commonly used to reorder retrieved chunks based on query similarity, naive similarity-based ordering often disrupts the original semantic flow of documents when small chunks are used, resulting in a fragmented and confusing context for generation. To mitigate this issue, we introduce contextual reranking, which reconstructs a coherent context by jointly considering relevance and document structure. After similarity-based retrieval, if multiple retrieved chunks originate from the same document and correspond to consecutive segments in the original text, we remove their overlapping spans and concatenate them into a single contiguous passage. This process not only restores the original narrative flow of the source document but also preserves the benefits of relevance-based reranking, enabling the model to consume structurally coherent and query-focused contexts. An illustrative example of this contextual reranking process is provided in Appendix B.

4 Experiments

4.1 Baselines

We compare our method against a comprehensive set of baselines, including a Pure LLM setting without retrieval, a Gold LLM that uses oracle gold pas-

Baselines	Multi Domain Dataset (Acc)			Multi Hop Dataset (F1)				Multi Hop Dataset (EM)			
	Open	Finance	Novel	HotpotQA	MultihopRAG	MuSiQue	2Wiki	HotpotQA	MultihopRAG	MuSiQue	2Wiki
pure LLM	0.490	0.210	0.075	0.331	0.175	0.149	0.181	0.247	0.175	0.090	0.161
Gold LLM	0.705	-	-	0.736	0.800	0.576	0.775	0.586	0.794	0.427	0.659
Naïve	0.642	0.342	0.282	0.678	0.701	0.428	0.594	0.525	0.692	0.302	0.509
RAPTOR (Sarthi et al.)	0.618	ERR	0.375	0.611	0.358	0.415	0.518	0.449	0.308	0.277	0.415
LightRAG (Guo et al., 2025)	0.383	0.364	0.093	0.418	0.544	0.187	0.270	0.315	0.538	0.100	0.235
HippoRAG2 (Gutiérrez et al.)	0.465	0.423	Token Limit	0.677	Token Limit	0.452	0.668	0.532	Token Limit	**0.325**	0.573
HyperGraphRAG (Luo et al., 2025)	0.255	ERR	0.054	0.562	0.606	0.285	0.379	0.423	0.592	0.175	0.339
ITER-RETGEN (Shao et al., 2023)	0.628	0.326	0.321	0.675	0.699	0.446	0.624	0.528	0.690	0.324	0.533
Self-Ask (Press et al., 2023)	0.598	0.318	0.300	0.669	0.628	0.370	0.582	0.522	0.623	0.255	0.498
Ours	**0.668**	**0.432**	**0.386**	**0.706**	**0.789**	**0.461**	**0.670**	**0.550**	**0.779**	**0.325**	**0.559**
w/o contextual reranking	0.634	0.422	0.360	0.705	0.756	0.454	0.665	0.541	0.747	0.310	0.542
w/o query expansion	0.650	0.428	0.375	0.705	0.775	0.446	0.666	0.547	0.765	0.311	0.554

Table 4: Results on multi-domain and multi-hop QA benchmarks. Accuracy (Acc) is reported for multi-domain QA, while F1 and Exact Match (EM) are reported for multi-hop QA. All results are evaluated under a fixed context token budget. "ERR" indicates that the method failed to run due to runtime errors, and "Token Limit" denotes cases where the retrieved context exceeded the model's maximum context length. Our method consistently outperforms all baselines across domains and reasoning benchmarks.

sages, and a Naïve RAG baseline based on standard dense retrieval. We further evaluate recent document transformation–based RAG approaches, covering both summarization-based methods such as RAPTOR (Sarthi et al.) and decomposition-based methods such as LightRAG (Guo et al., 2025), HippoRAG2 (Gutiérrez et al.), and HyperGraphRAG (Luo et al., 2025). In addition, we include iterative generation methods, namely Self-Ask (Press et al., 2023) and ITER-RETGEN (Shao et al., 2023), to compare their effectiveness and efficiency with our method on multi-hop reasoning tasks. For both iterative methods, we configure the generation process to repeat answer generation for up to three iterations, following prior work.

4.2 Datasets

To ensure fair evaluation across diverse retrieval scenarios, we conduct experiments on seven datasets covering multiple QA settings. These consist of one open-domain dataset (ClapNQ (Rosenthal et al., 2025)), two domain-specific datasets (FiQA (Maia et al., 2018) for finance and NovelQA (Wang et al., 2024) for long-form narrative documents), and four multi-hop reasoning benchmarks (HotpotQA (Yang et al., 2018), MultihopRAG (Tang and Yang), MuSiQue (Trivedi et al., 2022), and 2WikiMultiHopQA (Ho et al., 2020)). To evaluate both descriptive and factoid answering, we adopt long-form gold answers for ClapNQ, FiQA, and NovelQA following the RAGChecker (Ru et al., 2024) protocol, while short-form gold answers are used for all multi-hop datasets. To reflect realistic deployment conditions, we intentionally include large-scale corpora, where FiQA contains 57K documents in total and NovelQA consists of extremely long single documents with lengths reaching up to 374K tokens. Detailed dataset statistics are reported in Table 3.

4.3 Metrics

For datasets evaluated with long-form answers, we employ an LLM-as-a-judge (Zheng et al., 2023; Liu et al., 2023) framework that compares generated answers against gold answers and classifies them as correct or incorrect. For short-answer datasets, we follow the evaluation protocol of HippoRAG2 (Gutiérrez et al.) and report Exact Match (EM) and F1 scores. The prompt used for long-answer evaluation is provided in Appendix A.2 to ensure reproducibility.

4.4 Implementation Details

All RAG-based methods use GPT-4o-mini as the language model and text-embedding-small as the embedding model. Because different methods construct their final contexts using heterogeneous units, such as chunks, triplets, named entities, or atomic facts, using a single unified generation prompt would fail to fully utilize the provided context for certain approaches. Therefore, we adopt each method's original generation prompt when available, while Pure LLM, Naïve RAG, and our method use the prompt from HippoRAG2 (Gutiérrez et al.). To ensure a fair comparison, we constrain the final context length to 5,400 tokens for all methods, as discussed in Section 2.

4.5 Main Results

Table 4 reports the main results across three multi-domain QA datasets and four multi-hop QA benchmarks. Overall, our method consistently outperforms all baselines under a fixed context token budget, while remaining below the Gold LLM up-

per bound. On multi-domain QA, our approach achieves the highest accuracy across all evaluated domains, obtaining scores of 0.668 on Open-domain, 0.432 on Finance, and 0.386 on Novel. For multi-hop QA, our method yields clear and consistent improvements in both F1 and Exact Match, achieving 0.706 / 0.550 on HotpotQA, 0.789 / 0.779 on MultihopRAG, 0.461 / 0.325 on MuSiQue, and 0.670 / 0.559 on 2WikiMultiHopQA, thereby outperforming graph-based RAG variants and iterative generation approaches. Notably, several structured RAG methods fail under realistic settings due to excessive document length or corpus size, resulting in token limit violations or runtime errors, whereas our method remains stable across all datasets, highlighting its robustness under practical context constraints. These results confirm that recall-oriented retrieval combined with context-aware reconstruction is a key factor in improving RAG performance, particularly for complex multi-hop reasoning, without incurring the inefficiencies of LLM-heavy iterative generation.

The ablation results in Table 4 show that removing either contextual reranking or query expansion consistently degrades performance across both multi-domain and multi-hop benchmarks. In particular, eliminating query expansion leads to noticeable drops on multi-hop reasoning datasets, confirming its role in improving recall for compositional evidence chains, while removing contextual reranking reduces overall accuracy and stability by weakening relevance-aware context construction. These results indicate that both components are complementary and jointly necessary for the strong performance of the full model.

4.6 Multi-hop Efficiency

To assess whether an LLM-free Query Expansion strategy can efficiently handle complex multi-hop queries, we compare our method against iterative generation approaches, Self-Ask (Press et al., 2023) and ITER-RETGEN (Shao et al., 2023), focusing on both QA performance and retrieval cost. Table 5 reports Exact Match, F1, retrieval time, retrieval token cost, and iteration count across four multi-hop QA benchmarks.

Here, Single Query Expansion denotes performing query expansion once on the original query, while Double Query Expansion applies an additional expansion step to the already expanded query, resulting in two consecutive retrieval repetitions without intermediate generation. Across all datasets, Single Query Expansion consistently improves performance over the no-expansion baseline, achieving higher EM and F1 scores with only a modest increase in retrieval time and no additional token cost.

Extending expansion to Double Query Expansion yields diminishing or unstable gains while substantially increasing retrieval latency, indicating limited benefit from repeatedly expanding expanded queries. In contrast, iterative generation methods incur significantly higher latency and token consumption due to repeated LLM invocations across multiple retrieval–generation cycles.

Overall, these results suggest that a single, carefully designed LLM-free Query Expansion step is sufficient for most multi-hop reasoning scenarios, offering a favorable balance between effectiveness and efficiency without relying on iterative LLM calls.

5 Related Works

5.1 Document Transformation Approaches for Retrieval-Augmented Generation

As Retrieval-Augmented Generation (RAG) is applied to large-scale and long-form corpora such as financial, legal, and medical documents, naive retrieval of raw texts often exceeds practical context limits and incurs high computational cost. To address this, many studies introduce document transformation strategies that modify documents before or after retrieval to improve efficiency and downstream generation quality.

Representative methods include summarization-based RAG, which compresses long documents into concise representations to reduce token usage (Sarthi et al.; Achkar et al., 2025; Hong et al., 2025), as well as structural decomposition techniques that break documents into atomic facts, entities, or relational triplets to expose fine-grained evidence for reasoning (Guo et al., 2025; Gutiérrez et al.; Luo et al., 2025). Other work explores rewriting or paraphrasing to expand implicit information or align document content with query semantics (Ji et al., 2024; Xian et al., 2025), often combining multiple transformations to further increase retrieval flexibility and coverage (Edge et al., 2024). Despite their efficiency benefits, recent studies (Liu et al., 2024; Laitenberger et al., 2025) suggest that document transformation can introduce information loss and noise, particularly for multi-hop reasoning tasks that rely on precise factual dependencies.

	EM	F1	Retrieval time (sec)	Retrieval token cost	# Iteration	EM	F1	Retrieval time (sec)	Retrieval token cost	# Iteration
	HotpotQA					MultihopRAG				
w/o Query Expansion	0.547	0.705	39	0	0	0.765	0.775	38	0	0
Single Query Expansion	**0.550**	**0.706**	55	0	1	**0.779**	**0.789**	59	0	1
Double Query Expansion	**0.550**	**0.706**	101	0	2	0.758	0.767	103	0	2
Iter-RETGEN (Shao et al., 2023)	0.528	0.675	479	20341	3	0.690	0.699	510	21303.7	3
Self-Ask (Press et al., 2023)	0.522	0.669	4510	16439.2	2.98	0.623	0.628	4636	16524.5	2.97
	Musique					2wiki				
w/o Query Expansion	0.311	0.446	37	0	0	0.554	0.666	35	0	0
Single Query Expansion	**0.331**	**0.463**	55	0	1	**0.559**	**0.667**	48	0	1
Double Query Expansion	0.325	0.461	107	0	2	0.556	0.665	98	0	2
Iter-RETGEN (Shao et al., 2023)	0.324	0.446	521	20403.4	3	0.533	0.624	417	20364.4	3
Self-Ask (Press et al., 2023)	0.255	0.370	3967	16650.8	3	0.498	0.582	4957	16624.2	3

Table 5: Comparison of query expansion strategies and iterative generation methods on multi-hop QA benchmarks. Results include EM, F1, retrieval time, retrieval token cost, and iteration count across four multi-hop QA datasets. Iteration denotes the number of retrieval–generation cycles for Iter-RETGEN and Self-Ask, and the number of retrieval repetitions for our method. Token cost measures additional tokens consumed during retrieval and generation.

In this work, we empirically validate this limitation through a controlled analysis that isolates the effects of document transformation from those of context length and retrieval budget. Based on these findings, we propose a new RAG design principle that prioritizes maximizing retrieval recall under a fixed context length while minimizing transformation-induced noise, leading to a more efficient and reliable multi-hop RAG framework.

5.2 Iterative generation for Multihop QA

Multi-hop question answering requires models to integrate evidence distributed across multiple documents or reasoning steps, which has motivated a line of research on iterative generation–based RAG pipelines. Early approaches (Press et al., 2023) explicitly decompose a complex query into a sequence of intermediate sub-questions, retrieve evidence for each step, and iteratively refine the answer. Subsequent work (Shao et al., 2023) extends this paradigm by alternating between generation and retrieval, where intermediate answers or rationales are used as new queries to retrieve additional evidence.

These methods aim to emulate step-by-step human reasoning and have shown effectiveness on multi-hop benchmarks by progressively expanding the evidence set. However, such pipelines typically require multiple LLM invocations per query, tightly coupling retrieval with generation. As a result, they introduce substantial latency, token cost, and instability, particularly when intermediate generations are noisy or partially incorrect, which can propagate errors to subsequent steps.

In this work, we revisit multi-hop RAG from the perspective that high-recall retrieval, rather than LLM-driven iterative reasoning, can reduce or even eliminate the need for iterative generation.
We provide additional discussion of query expansion and PRF in Appendix C.

6 Conclusion

In this paper, we conducted a systematic empirical study of retrieval and context construction strategies in Retrieval-Augmented Generation (RAG), focusing on the effects of chunk size, context length, and document transformation. Our analysis led to three main findings: first, top-k-based evaluation is highly sensitive to chunk granularity and fails to reflect the actual information budget used during generation; second, document transformation strategies consistently introduce information loss and noise, which is particularly harmful for multi-hop reasoning; and third, retrieval recall under a fixed context length is the primary driver of generation quality. Based on these findings, we proposed a recall-oriented RAG framework that combines fine-grained chunking, an LLM-free PRF-style Query Expansion, and contextual reranking. This design eliminates iterative LLM calls while maintaining high evidence coverage. Experimental results across seven datasets demonstrate that the proposed approach consistently outperforms strong baselines while remaining robust to large corpora and long documents, and efficient for multi-hop reasoning tasks.

Limitations

Despite the strong empirical results, our work has several limitations that warrant further investigation.

First, our LLM-free Query Expansion strategy is designed to be lightweight and efficient, but it relies on pseudo-relevance feedback rather than adaptive reasoning signals. Although a single expansion step improves recall in the multi-hop settings studied here, the method may be less reliable for queries requiring long sequential inference chains, such as cases involving more than four reasoning steps. Repeated feedback-based expansion can amplify early retrieval errors and gradually shift the expanded query away from the original information need, resulting in query drift, a known trade-off in PRF and feedback-based retrieval methods. This is also reflected in our experiments, where Double Query Expansion provides diminishing or unstable gains while substantially increasing retrieval latency. Future work should explore adaptive or selective feedback mechanisms, such as topic-aware or quality-aware PRF variants. These mechanisms could determine whether feedback chunks are sufficiently reliable before expansion, thereby extending our framework to more complex reasoning scenarios while reducing the risk of query drift.

Second, our evaluation focuses on accuracy-oriented metrics such as EM, F1, and LLM-as-a-judge correctness. While these metrics capture answer quality, they do not fully reflect broader user-centric factors such as faithfulness attribution, interpretability of retrieved evidence, or robustness under adversarial or noisy queries. Incorporating richer evaluation dimensions remains an important avenue for future work.

References

Pierre Achkar, Tim Gollub, and Martin Potthast. 2025. Ask, retrieve, summarize: A modular pipeline for scientific literature summarization. *arXiv preprint arXiv:2505.16349*.

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

Zirui Guo, Lianghao Xia, Yanhua Yu, Tu Ao, and Chao Huang. 2025. LightRAG: Simple and fast retrieval-augmented generation. In *Findings of the Association for Computational Linguistics: EMNLP 2025*, pages 10746–10761, Suzhou, China. Association for Computational Linguistics.

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

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

Yubin Hong, Chaofan Li, Jingyi Zhang, and Yingxia Shao. 2025. Fg-rag: Enhancing query-focused summarization with context-aware fine-grained graph rag. *ArXiv*.

Lei Huang, Weijiang Yu, Weitao Ma, Weihong Zhong, Zhangyin Feng, Haotian Wang, Qianglong Chen, Weihua Peng, Xiaocheng Feng, Bing Qin, and 1 others. 2025. A survey on hallucination in large language models: Principles, taxonomy, challenges, and open questions. *ACM Transactions on Information Systems*, 43(2):1–55.

Yuelyu Ji, Zhuochun Li, Rui Meng, Sonish Sivarajkumar, Yanshan Wang, Zeshui Yu, Hui Ji, Yushui Han, Hanyu Zeng, and Daqing He. 2024. RAG-RLRC-LaySum at BioLaySumm: Integrating retrieval-augmented generation and readability control for layman summarization of biomedical texts. In *Proceedings of the 23rd Workshop on Biomedical Natural Language Processing*, pages 810–817, Bangkok, Thailand. Association for Computational Linguistics.

Alex Laitenberger, Christopher D Manning, and Nelson F. Liu. 2025. Stronger baselines for retrieval-augmented generation with long-context language models. In *Proceedings of the 2025 Conference on Empirical Methods in Natural Language Processing*, pages 32559–32569, Suzhou, China. Association for Computational Linguistics.

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

Xinxin Liu. 2024. A survey of hallucination problems based on large language models. *Applied and Computational Engineering*, 97:24–30.

Yang Liu, Dan Iter, Yichong Xu, Shuohang Wang, Ruochen Xu, and Chenguang Zhu. 2023. G-eval: NLG evaluation using gpt-4 with better human alignment. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 2511–2522, Singapore. Association for Computational Linguistics.

Haoran Luo, Guanting Chen, Yandan Zheng, Xiaobao Wu, Yikai Guo, Qika Lin, Yu Feng, Zemin Kuang, Meina Song, Yifan Zhu, and 1 others. 2025. Hypergraphrag: Retrieval-augmented generation via hypergraph-structured knowledge representation. *arXiv preprint arXiv:2503.21322*.

Macedo Maia, Siegfried Handschuh, André Freitas, Brian Davis, Ross McDermott, Manel Zarrouk, and Alexandra Balahur. 2018. Www'18 open challenge: financial opinion mining and question answering. In *Companion proceedings of the the web conference 2018*, pages 1941–1942.

Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah A Smith, and Mike Lewis. 2023. Measuring and narrowing the compositionality gap in language models. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 5687–5711.

Sara Rosenthal, Avirup Sil, Radu Florian, and Salim Roukos. 2025. Clapnq: C ohesive l ong-form a nswers from p assages in natural questions for rag systems. *Transactions of the Association for Computational Linguistics*, 13:53–72.

Dongyu Ru, Lin Qiu, Xiangkun Hu, Tianhang Zhang, Peng Shi, Shuaichen Chang, Cheng Jiayang, Cunxiang Wang, Shichao Sun, Huanyu Li, and 1 others. 2024. Ragchecker: A fine-grained framework for diagnosing retrieval-augmented generation. *Advances in Neural Information Processing Systems*, 37:21999–22027.

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*.

Zhihong Shao, Yeyun Gong, Yelong Shen, Minlie Huang, Nan Duan, and Weizhu Chen. 2023. Enhancing retrieval-augmented large language models with iterative retrieval-generation synergy. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pages 9248–9274.

Yixuan Tang and Yi Yang. Multihop-rag: Benchmarking retrieval-augmented generation for multi-hop queries. In *First Conference on Language Modeling*.

SM Towhidul Islam Tonmoy, SM Mehedi Zaman, Vinija Jain, Anku Rani, Vipula Rawte, Aman Chadha, and Amitava Das. 2024. A comprehensive survey of hallucination mitigation techniques in large language models. *CoRR*.

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

Cunxiang Wang, Ruoxi Ning, Boqi Pan, Tonghui Wu, Qipeng Guo, Cheng Deng, Guangsheng Bao, Qian Wang, and Yue Zhang. 2024. Novelqa: A benchmark for long-range novel question answering. *CoRR*.

Xun Xian, Ganghua Wang, Xuan Bi, Jayanth Srinivasa, Ashish Kundu, Charles Fleming, Mingyi Hong, and Jie Ding. 2025. On the vulnerability of applying retrieval-augmented generation within knowledge-intensive application domains. In *International Conference on Learning Representations (ICLR) 2025*. Accepted at ICLR 2025.

Weiye Xu, Min Wang, Wengang Zhou, and Houqiang Li. 2024. P-rag: Progressive retrieval augmented generation for planning on embodied everyday task. In *Proceedings of the 32nd ACM International Conference on Multimedia*, pages 6969–6978.

Xiao Yang, Kai Sun, Hao Xin, Yushi Sun, Nikita Bhalla, Xiangsen Chen, Sajal Choudhary, Rongze Daniel Gui, Ziran Will Jiang, Ziyu Jiang, Lingkun Kong, Brian Moran, Jiaqi Wang, Yifan Ethan Xu, An Yan, Chenyu Yang, Eting Yuan, Hanwen Zha, Nan Tang, and 8 others. 2024. Crag - comprehensive rag benchmark. In *Advances in Neural Information Processing Systems*, volume 37, pages 10470–10490. Curran Associates, Inc.

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

Yue Zhang, Yafu Li, Leyang Cui, Deng Cai, Lemao Liu, Tingchen Fu, Xinting Huang, Enbo Zhao, Yu Zhang, Yulong Chen, and 1 others. 2025. siren's song in the ai ocean: A survey on hallucination in large language models. *Computational Linguistics*, pages 1–46.

Lianmin Zheng, Wei-Lin Chiang, Ying Sheng, Siyuan Zhuang, Zhanghao Wu, Yonghao Zhuang, Zi Lin, Zhuohan Li, Dacheng Li, Eric Xing, and 1 others. 2023. Judging llm-as-a-judge with mt-bench and chatbot arena. *Advances in neural information processing systems*, 36:46595–46623.

A Overview of the used prompts

A.1 Document Decomposition Prompt

Atomic Decomposition Prompt

`You are an information extraction system.`

Your task is to extract atomic facts from a given text chunk.

Definition:
An atomic fact is a single, concise, and independently verifiable factual claim.

Each atomic fact must:
– Contain exactly ONE factual claim
– Be understandable without additional context
– Be verifiable against an external knowledge source
– Avoid opinions, interpretations, or explanations
– Not combine multiple facts into one sentence

If a sentence contains multiple facts, split them into multiple atomic facts.
If a sentence contains no verifiable factual claim, ignore it.

Return the result in JSON format only, following the structure shown in the example.
Do NOT include numbering or any additional text outside the JSON.

—

Example

Chunk:
Marie Curie was born in Warsaw in 1867. She discovered polonium and radium.

Output:

```
{
  "atomic_facts": [
    "Marie Curie was born in Warsaw.",
    "Marie Curie was born in 1867.",
    "Marie Curie discovered polonium.",
    "Marie Curie discovered radium."
  ]
}
```

Table 6: Prompt used for atomic fact decomposition.

A.2 Answer Evaluation Prompt

Answer Evaluation Prompt

`You are an expert evaluator for question-answering systems.`
Your task is to determine if a predicted answer is correct given a question and the gold (correct) answer.

Evaluation Guidelines:
1. The predicted answer is CORRECT if it contains the essential information from the gold answer, even if worded differently.
2. The predicted answer is CORRECT if it provides the same factual information.
3. The predicted answer is INCORRECT if it contradicts the gold answer or provides wrong information.
4. The predicted answer is INCORRECT if it contains too much additional information compared with the gold answer.
5. Minor differences in wording, formatting, or additional context are acceptable as long as the core fact is correct.

Your Task:
Evaluate whether the predicted answer is correct or incorrect.
Respond with ONLY a JSON object in the following exact format:

```
{
  "verdict": "CORRECT" or "INCORRECT",
  "confidence": "HIGH" or "MEDIUM" or "LOW",
  "reason": "Brief explanation of your decision (1–2 sentences)"
}
```

Do not include any text before or after the JSON object.

Table 7: Prompt used for answer correctness evaluation in Open-Ended QA.

B Method illustrations

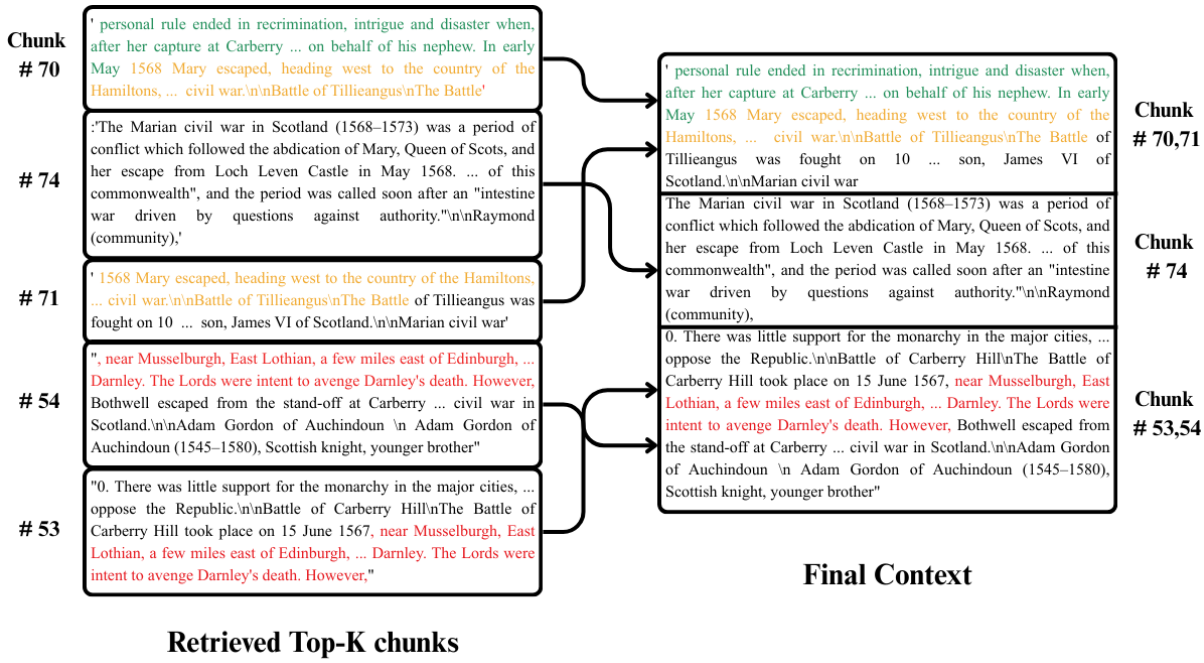


Figure 4: Overview of contextual reranking. Consecutive retrieved chunks from the same document are merged to reconstruct coherent passages, preserving both the original narrative flow and the relevance gains of similarity-based reranking.

C Query Expansion and Pseudo-Relevance Feedback

Query expansion has long been studied in information retrieval as a way to address vocabulary mismatch and improve recall. A representative approach is pseudo-relevance feedback (PRF), which assumes that top-ranked results from an initial retrieval pass are relevant and uses them to enrich the original query. Classical PRF methods, such as Rocchio-style feedback and relevance models, expand or reweight the query based on terms from initially retrieved documents. Recent dense retrieval studies further show that PRF can improve query representations by incorporating feedback from top-ranked passages.

Our method follows the same high-level principle but adapts PRF to retrieval-augmented generation under a fixed context budget. Instead of expanding the query with generated sub-questions or transformed document summaries, we use raw fine-grained chunks from the first retrieval pass as pseudo-relevant evidence. This enables recall-oriented multi-hop retrieval without additional LLM calls, while avoiding the information loss caused by document transformation.

D Discussion

D.1 Faithfulness and Evidence Grounding under Recall-Oriented Retrieval

A potential concern with recall-oriented retrieval is that maximizing retrieval recall may introduce additional irrelevant or weakly related evidence into the final context. In principle, such noisy evidence could affect faithfulness if the language model relies on incorrect, misleading, or only superficially related information during generation. This raises an important question: does improving recall necessarily come at the cost of evidence grounding quality?

Our empirical results suggest that this trade-off is not inherent under realistic fixed context budgets. Across our controlled experiments, answer accuracy improves most consistently when the correct supporting evidence is included in the context, even when the context also contains some irrelevant chunks. This indicates that modern large language models are generally capable of filtering out moderate amounts of retrieval noise during generation, whereas the absence, omission, or distortion of necessary evidence is substantially more harmful. In other words, irrelevant evidence is less damaging than missing evidence, especially in multi-hop reasoning settings where each reasoning step depends on the availability of specific supporting facts.

This observation does not imply that retrieval noise is harmless. Rather, it suggests that recall-oriented retrieval and faithfulness are not inherently in conflict when context construction is carefully controlled. Our framework therefore prioritizes factual completeness first, while reducing the risk of noisy grounding through contextual reranking and budget-aware context construction. By preserving raw evidence and reconstructing locally coherent passages, our method aims to provide the model with sufficient supporting information without relying on aggressive document transformations that may weaken factual grounding.

Future work should further evaluate this relationship using explicit faithfulness and attribution metrics, such as whether generated claims are directly supported by retrieved evidence. Nevertheless, the current results support the view that preserving complete evidence within a realistic context budget is a practical foundation for both accurate and faithful RAG generation.

D.2 Evaluation under Native Baseline Configurations

In the main experiments, we evaluate all methods under a fixed context token budget in order to isolate the effect of retrieval and context construction from differences in context length. While this controlled setting enables fair comparison across heterogeneous RAG systems, some baselines were originally designed with their own retrieval strategies, context construction procedures, and generation prompts. To examine whether our conclusions depend on the fixed-budget evaluation protocol, we additionally conduct experiments in which major baselines are evaluated under their native configurations.

Due to computational constraints, we perform these additional experiments on representative datasets, including the open-domain NQ dataset and three multi-hop reasoning benchmarks: HotpotQA, MuSiQue, and 2WikiMultiHopQA. In this setting, each baseline is run using its original retrieval strategy, context construction method, and generation prompt whenever available. The results are reported in Table 8.

The results show that our recall-oriented method continues to consistently outperform structured transformation-based and iterative baselines across

Model	Open (Acc)	HotpotQA (F1)	MuSiQue (F1)	2Wiki (F1)
Pure LLM	0.490	0.331	0.149	0.181
Gold LLM	0.705	0.736	0.576	0.775
Naïve	0.642	0.678	0.428	0.594
RAPTOR	0.618	0.611	0.415	0.518
LightRAG	0.465	0.421	0.217	0.404
HippoRAG2	0.433	0.663	0.385	0.559
HyperGraphRAG	0.374	0.661	0.360	0.499
ITER-RETGEN	0.575	0.577	0.221	0.385
Self-Ask	0.535	0.647	0.293	0.528
Ours (recall-oriented)	**0.668**	**0.706**	**0.461**	**0.667**

Table 8: Results under native baseline configurations. Major baselines are evaluated using their original retrieval strategies, context construction methods, and generation prompts. Accuracy is reported for the open-domain dataset, while F1 is reported for multi-hop benchmarks.

both open-domain and multi-hop benchmarks. In particular, the performance gains on MuSiQue and 2WikiMultiHopQA remain stable under native configurations, indicating that our findings are not artifacts of restrictive token limits. Instead, they reflect a systematic trend: methods that preserve and retrieve sufficient supporting evidence tend to be more effective than methods that heavily transform, compress, or iteratively regenerate intermediate evidence.

These additional experiments further support our main conclusion that retrieval effectiveness is primarily determined by the preservation of relevant evidence within the final context. Although native configurations may allow individual baselines to use longer contexts or method-specific prompts, they do not eliminate the information loss and grounding weaknesses introduced by aggressive document transformation. We therefore find that the central observations of our study remain consistent across both controlled fixed-budget evaluation and more method-specific native evaluation settings.

D.3 Hyperparameter Analysis

We further analyze the sensitivity of our method to two key hyperparameters: the chunk size used for retrieval and the number of top-ranked feedback chunks used in query expansion. Specifically, we vary both chunk size and Top-K for query expansion, and evaluate their effects on HotpotQA and ClapNQ.

Across both datasets, we observe consistent trends with respect to chunk granularity and Top-K selection. First, smaller chunk sizes, particularly 150–300 tokens, consistently achieve the best performance across different Top-K settings. As the chunk size increases, performance gradually decreases, regardless of how many expansion candidates are used. This supports our main finding that fine-grained chunking is important for maximizing evidence coverage under a fixed context budget.

HotpotQA (F1 Score)				
Chunk Size	**Top-5**	**Top-10**	**Top-20**	**Top-40**
150 tokens	**0.706**	0.695	0.690	0.690
300 tokens	0.695	0.705	0.702	0.702
450 tokens	0.700	0.694	0.694	0.692
600 tokens	0.682	0.678	0.677	0.677
900 tokens	0.683	0.680	0.675	0.680
1200 tokens	0.679	0.668	0.668	0.653
ClapNQ (Accuracy)				
Chunk Size	**Top-5**	**Top-10**	**Top-20**	**Top-40**
150 tokens	**0.668**	0.665	0.665	0.659
300 tokens	0.650	0.664	0.658	0.658
450 tokens	0.652	0.652	0.641	0.639
600 tokens	0.623	0.637	0.621	0.621
900 tokens	0.637	0.628	0.623	0.625
1200 tokens	0.607	0.599	0.598	0.598

Table 9: Hyperparameter analysis of chunk size and Top-K for query expansion on HotpotQA and ClapNQ. HotpotQA is evaluated using F1 score, while ClapNQ is evaluated using accuracy.

Second, increasing Top-K in the query expansion step does not lead to systematic improvements. On both HotpotQA and ClapNQ, Top-5 and Top-10 generally achieve the best or near-best performance, while further increasing Top-K to 20 or 40 yields only marginal gains or even slight performance drops. This suggests that once the most relevant feedback chunks are included, additional lower-ranked chunks are more likely to introduce weakly related or noisy evidence than to provide useful supporting information.

Moreover, the performance variations caused by different Top-K values are relatively small compared to those caused by changes in chunk size.

Method	Time (s)	# LLM Calls
Naïve	13.8	0
RAPTOR	37.2	0
LightRAG	1671	1
HippoRAG2	1483	1
HyperGraphRAG	1087	1
ITER-RETGEN	1688	3
Self-Ask	2250	2.98
Ours	**29.5**	**0**

Table 10: End-to-end retrieval-stage efficiency on HotpotQA under each method's original configuration. The reported time is measured over 1,000 queries and includes retrieval, reranking, intermediate reasoning, and other retrieval-stage processing, but excludes final answer generation.

This indicates that our method is robust to moderate variations in Top-K, while chunk granularity remains the dominant factor affecting retrieval quality and downstream answer accuracy.

Overall, these results justify our design choice of using fine-grained chunks with a small Top-K for LLM-free query expansion. The observed trends are consistent with our main experimental findings and further demonstrate that aggressively increasing the number of expansion candidates does not provide meaningful benefits under realistic context constraints.

D.4 Efficiency under Original Baseline Configurations

To further evaluate the practical efficiency of different RAG pipelines, we conduct an additional experiment on the HotpotQA dataset, which requires multi-hop reasoning. For each method, we measure the total processing time per 1,000 queries. Unlike the fixed-budget comparison in the main experiments, all baselines are evaluated under their original configurations and implementation settings, including their native retrieval strategies, context construction procedures, and intermediate reasoning modules.

The reported time includes all retrieval-stage operations, such as index lookup, retrieval, reranking, and intermediate generation steps used during retrieval or reasoning. We exclude the final answer generation step in order to isolate the computational cost introduced by each retrieval and reasoning pipeline. We also report the number of LLM calls required during the retrieval and reasoning stage.

As shown in Table 10, our method achieves substantially lower latency than structured and LLM-dependent baselines while requiring no retrieval-time LLM calls. Although Naïve retrieval is faster due to its minimal retrieval pipeline, our method remains lightweight and provides stronger multi-hop reasoning performance through LLM-free query expansion and contextual reranking.

The observed efficiency differences can be explained by the structural design of each method. RAPTOR constructs a hierarchical summary tree and performs recursive traversal from higher-level summaries to leaf nodes. As a result, retrieval is repeated along the tree depth, leading to increased latency compared to simple dense retrieval.

LightRAG and HyperGraphRAG perform LLM-based keyword extraction or structural construction steps. These preprocessing and reasoning stages substantially increase both runtime and token consumption. HippoRAG2 applies LLM-based fact scoring and reranking over retrieved facts. Although it performs only a single LLM call during retrieval, this step introduces significant additional latency compared to purely embedding-based reranking.

ITER-RETGEN and Self-Ask explicitly rely on iterative retrieval–generation loops. They repeatedly generate intermediate answers, rationales, or sub-queries using LLMs and then perform additional retrieval steps. Consequently, their runtime and token cost scale with the number of iterations, resulting in the highest latency among all methods.

In contrast, our method performs query expansion using similarity-based retrieval without invoking any LLM during the retrieval stage. All expansion, reranking, and context construction steps are based on embedding similarity and lightweight post-processing. As a result, our approach avoids both iterative generation overhead and document transformation overhead, leading to low latency and zero additional retrieval-time token cost.

Overall, these results confirm that the proposed "LLM-free query expansion without extra cost" is not only a conceptual advantage but also a practical one. Our method achieves an order-of-magnitude reduction in retrieval-time latency compared to LLM-dependent baselines, while preserving strong answer quality.

Method	HotpotQA	MultihopRAG	MuSiQue	2Wiki
Naïve RAG	0.739	0.838	0.430	0.612
RAPTOR	0.629	0.596	0.423	0.554
LightRAG	0.508	0.680	0.265	0.443
HippoRAG2	0.781	Token Limit	0.431	0.646
HyperGraphRAG	0.776	0.843	0.459	0.566
ITER-RETGEN	0.670	0.792	0.312	0.587
Self-Ask	0.710	0.781	0.239	0.422
Ours	**0.836**	**0.881**	**0.507**	**0.720**

Table 11: LLM-as-a-judge accuracy on multi-hop QA datasets. Compared with EM and F1, this metric accounts for semantic equivalence, paraphrased answers, and minor wording variations.

E Additional LLM-based Evaluation on Multi-hop QA

Exact Match (EM) and token-level F1 are widely used for evaluating short-answer multi-hop question answering. However, LLM-generated answers may contain additional explanatory phrases, paraphrased expressions, or minor formatting variations even when the core answer is factually correct. In such cases, EM and F1 can underestimate answer quality because they are sensitive to surface-form differences.

To address this concern, we additionally evaluate the multi-hop QA datasets using an LLM-as-a-judge accuracy metric. Specifically, we use the same answer correctness evaluation prompt employed for long-answer tasks in our multi-domain experiments, which judges whether the predicted answer contains the essential information from the gold answer while allowing minor wording or formatting differences.

The results are reported in Table 11. Overall, LLM-based accuracy scores are generally higher than EM and F1 scores across methods. This confirms that EM and F1 tend to penalize answers that include correct information but differ from the gold answer in surface form or contain additional explanatory content. Several baseline methods benefit noticeably from this evaluation, suggesting that part of their lower EM/F1 scores is attributable to formatting or verbosity rather than factual errors.

Importantly, our method consistently achieves the best performance across all multi-hop datasets under this alternative metric as well. Our method obtains 0.836 on HotpotQA, 0.881 on MultihopRAG, 0.507 on MuSiQue, and 0.720 on 2WikiMultiHopQA. These results demonstrate that the superiority of our approach is robust across evaluation protocols and is not an artifact of exact string matching. Overall, the additional LLM-based evaluation further supports our main conclusion that recall-oriented retrieval under a fixed context budget improves answer quality in multi-hop RAG.

Artifact	Usage	License / Terms
ClapNQ	Evaluation dataset	Apache-2.0
FiQA	Evaluation dataset	dataset-specific terms / non-commercial
NovelQA	Evaluation dataset	Apache-2.0 + dataset-specific terms
HotpotQA	Evaluation dataset	CC BY-SA 4.0
MultihopRAG	Evaluation dataset	ODC-BY
MuSiQue	Evaluation dataset	CC BY 4.0
2WikiMultiHopQA	Evaluation dataset	Apache-2.0
Baselines	Comparative experiments	MIT

Table 12: Summary of artifacts used or released in this work.

F Artifact Licenses and Terms of Use.

We release our code for research and reproducibility purposes under the MIT License. We use publicly available benchmark datasets in accordance with their respective licenses and terms of use, including ClapNQ, FiQA, NovelQA, HotpotQA, MultihopRAG, MuSiQue, and 2WikiMultiHopQA. We do not redistribute the original datasets; instead, we provide scripts and instructions for obtaining them from their official sources. For baseline methods, we use official implementations when available and follow their corresponding licenses and usage terms. We use GPT-4o-mini and text-embedding-small through the official API in accordance with the provider's terms of use, and we do not redistribute proprietary model weights. A summary of the artifacts used or released in this work and their corresponding licenses or terms of use is provided in Table 12.