Haystack Engineering: Context Engineering Meets the Long-Context Challenge in Large Language Models

Anonymous Author(s)
Affiliation
Address
email

Abstract

Existing "needle-in-a-haystack" (NIAH) benchmarks for long-context LLM evaluation often overlook "context engineering", using random distractors rather than biased outputs of retrieval systems. We present HaystackCraft, a new NIAH benchmark built on the full English Wikipedia hyperlink network, which evaluates LLMs against ranked distractors from sparse, dense, hybrid, and graph-based retrievers. Experiments on 10 LLMs show significant performance degradation as context size increases. We find that distractor composition is crucial: semantically similar documents are more challenging than lexically similar ones. Graph-based reranking mitigates harmful distractors, improving the LLM performance by up to 44%.

1 Introduction

Long-context reasoning is fundamental for large language models (LLMs). Recent innovations have driven significant progress in this area (Su et al., 2024; Peng et al., 2024; Dao et al., 2022; Xiao et al., 2024). Consequently, modern LLMs can often achieve near-perfect recall on synthetic "needle-in-a-haystack" (NIAH) benchmarks (Yen et al., 2025), which test whether a model can retrieve relevant information (i.e., *needle*) from a large context that contains many distractors (i.e., *haystack*).

However, these successes can be misleading as they overlook "context engineering" (Mei et al., 2025), i.e., the practice of selecting and structuring information for an LLM's context. In practical applications like retrieval-augmented generation (RAG) (Lewis et al., 2020), distractors are not independent random samples, but ranked outputs of imperfect and biased retrieval systems. For instance, a sparse retriever populates the haystack with documents that are lexically similar but potentially semantically irrelevant (Robertson et al., 1994; Robertson & Zaragoza, 2009), while a dense retriever may return semantically related but factually incorrect "near misses" (Karpukhin et al., 2020). It is therefore essential to consider a representative set of heterogeneous retrievers. Furthermore, for complex multi-hop queries, needle documents are often interconnected within a larger document graph (e.g., webpage hyperlink networks). Graph-based retrieval methods are central to information retrieval and search engines (Page et al., 1999).

To systematically study the impact of context engineering on long-context reasoning, we introduce **haystack engineering**: the principled construction of noisy contexts using heterogeneous retrieval strategies. We explore this concept through **HaystackCraft**, our newly proposed NIAH benchmark built on the full English Wikipedia hyperlink network. HaystackCraft systematically examines how different retriever choices shape the distractor composition, haystack ordering, and the LLM performance. It evaluates a broad spectrum of widely adopted retrievers, including sparse, dense, hybrid, and graph-based methods. Previous NIAH benchmarks mostly consider query- and retriever-independent distractors (Kamradt; Yuan et al., 2024; Hsieh et al., 2024; Kuratov et al., 2024). While HELMET (Yen et al., 2025) employs a dense retriever for distractor construction, it does not address retriever heterogeneity, network-structured corpora, or the role of retriever-ranked ordering.

Our experiments on 10 long-context LLMs yield several key insights. We find that all models suffer a significant performance drop as context size increases to $128K$ tokens, with decreases ranging from 7.6% to 61.8%. Semantically similar distractors from dense retrievers are more challenging than lexically similar distractors from sparse retrievers. Furthermore, we observe that graph-based reranking using Personalized PageRank (PPR) substantially mitigates harmful distractors and improves performance across all models and base retrievers, particularly for multi-hop questions and at larger context sizes, with improvements as high as 44%. Finally, our analysis shows that haystack ordering has a complex, model-dependent impact, underscoring the importance of evaluating LLMs under retriever-ranked orders that reflect practical RAG systems.

2 HaystackCraft

2.1 A Framework for Haystack Engineering

We formalize the NIAH problem through the lens of context engineering to study distractor composition and haystack ordering. Let $\mathcal{D}$ be a document corpus. Given a query q, $\mathcal{N}_q \subset \mathcal{D}$ denotes the set of ground-truth documents required to correctly answer q, which we term the **needle**. A retriever $\mathcal{R}$ assigns each document $d \in \mathcal{D}$ a relevance score $\mathcal{R}(q, d) \in \mathbb{R}$, where a larger value indicates a higher relevance, thereby inducing a ranking of the documents. Given a target context size of S tokens, we construct the **haystack** set $\mathcal{H}_q^{\mathcal{R}}(S)$ by including all needles $\mathcal{N}_q$ and then filling the remaining token budget with the top-ranked distractors from $\mathcal{D} \setminus \mathcal{N}_q$. Finally, $\mathcal{H}_q^{\mathcal{R}}(S)$ is linearized into a sequence by an ordering policy $\pi(q, \mathcal{R}, \mathcal{H}_q^{\mathcal{R}}(S)) = (d_1, \cdots, d_{|\mathcal{H}_q^{\mathcal{R}}(S)|})$ for LLM consumption.

Retriever Choice ($\mathcal{R}$). The retriever choice is the primary mechanism for engineering the haystack's composition. HaystackCraft incorporates a broad spectrum of retrievers. 1) **Sparse Retriever** (i.e., BM25 (Robertson et al., 1994; Robertson & Zaragoza, 2009)): a classical retriever that measures lexical similarity. 2) **Dense Retriever** (i.e., Qwen3-Embedding-0.6B (Zhang et al., 2025)): a retriever that captures semantic similarity. We choose it in light of its competitive retrieval performance on MMTEB (Enevoldsen et al., 2025), small size, and applicability to long documents. 3) **Hybrid Retriever** (i.e., BM25 + Qwen3-Embedding-0.6B): a combination of the two using reciprocal rank fusion (Cormack et al., 2009), which is robust to score magnitude differences across retrievers and often yields better performance (Lee et al., 2023).

Graph-Based Retrieval for Multi-Hop Question Answering (QA). For complex multi-hop questions where needles are interconnected, standard retrievers fall short as they ignore inter-doc structures, which can offer strong retrieval cues. For instance, PageRank (Page et al., 1999), a foundational algorithm for modern search engines, leverages this by considering a document structurally important if it is heavily referenced by other important documents. Building on this idea, we employ Personalized PageRank (PPR) (Haveliwala, 2002) to study the impact of graph-based retrieval on distractor composition and downstream LLM performance. Specifically, we first use one of the three base retrievers above, then perform PPR reranking seeded on the top-N documents.

Haystack Ordering (π). LLMs exhibit strong positional biases, and the order of documents can significantly impact their long-context performance (Liu et al., 2024; Xiao et al., 2024; Yang et al., 2025c). While prior NIAH benchmarks often use random permutations to analyze this bias, practical RAG systems present documents in a ranked order determined by the retriever. To bridge this gap, we evaluate both retriever-ranked ordering and random permutations. This dual approach allows us to assess LLM performance in a realistic RAG setting while also isolating the effects of positional bias.

2.2 Corpus and QA samples

Networked Corpus. We employ the 2025-04-04 English Wikipedia dump as a unified corpus for both needles and distractors, which comprises $6,954,909$ articles interconnected by $97,442,472$ unique hyperlinks. We use full Wikipedia articles as the unit of retrieval, rather than smaller chunks, to preserve document integrity and present a more realistic long-context reasoning challenge.

QA Datasets. We use Natural Questions (NQ) (Kwiatkowski et al., 2019) for single-hop questions and MuSiQue (Trivedi et al., 2022) for multi-hop questions. Since both NQ and MuSiQue were created using older Wikipedia versions, we manually filter the samples to ensure validity against our

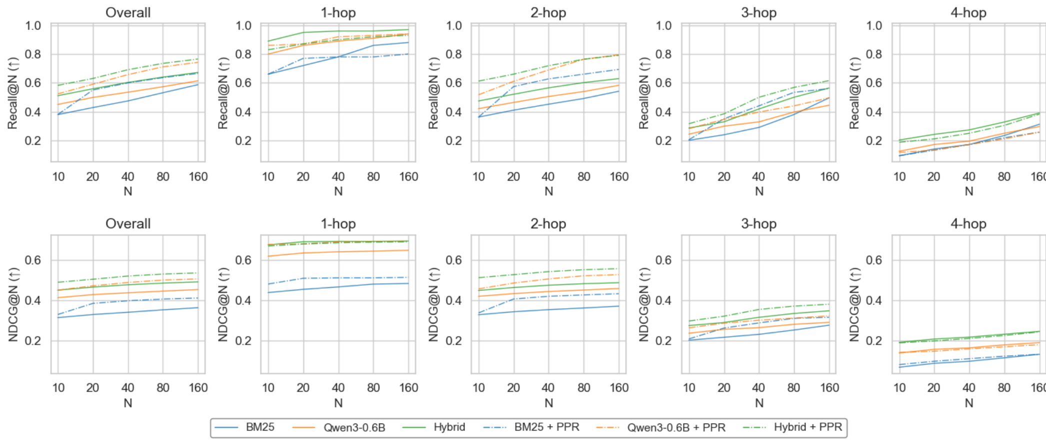

Figure 1: Evaluation of retrievers as the number of retrieved documents (N) increases.

updated corpus, yielding a final set of 500 high-quality samples where answers are unambiguous and fully grounded in the text. See Appendix A for further details like data contamination discussions.

3 Experiments

We evaluate 10 widely used long-context LLMs, including thinking models (Qwen3-14B (Yang et al., 2025b), Gemini 2.5 Flash-Lite, and o4-mini) and general-purpose models, such as GPT-4.1 mini and the open-source Llama-3.1 (Dubey et al., 2024), Qwen2.5-1M (Yang et al., 2025a), and Gemma 3 (Kamath et al., 2025) families. We evaluate each model across input context sizes of $S \in \{8K, 16K, 32K, 64K, 128K\}$. For more details, see Appendix B.

Retrieval Effectiveness. To ensure distractor quality, we first evaluate retriever effectiveness using NDCG $@N$ (Järvelin & Kekäläinen, 2000, 2002) in addition to Recall $@N$ to account for retrievers' ranking performance. As NIAH scales the number of distractors for long-context study, we study the scaling behaviors of the retrievers by gradually increasing N, the number of retrieved documents.

Fig. 1 presents the evaluation results. Among the base retrievers, the dense retriever (Qwen3-0.6B) consistently outperforms the sparse retriever (BM25) in both metrics, and combining them with a hybrid retriever further improves the performance. The retrieval effectiveness decreases as the question hop increases. Graph-based reranking substantially boosts all base retrievers, especially for multi-hop questions. Importantly, the retrieval performance exhibits nice scaling properties and continues to improve as N increases, without noticeable troublesome pattern shifts.

Impact of Retriever Choice. To holistically study the impact of retriever choice on haystack composition and ordering, we first employ retriever-ranked haystack ordering. Fig. 2 presents the evaluation results. All LLMs exhibit a substantial performance degradation as the context size extends to $128K$ tokens, with performance drops ranging from 7.6% to 61.8%. For larger context sizes, distractors constructed by the dense retriever (Qwen3-0.6B) based on semantic similarity are generally more challenging for the models than the lexical distractors from the sparse retriever (BM25). This is evidenced by additional performance drops of up to 9.6% (Llama-3.1-8B-Instruct) when faced with semantic distractors. Interestingly, the use of a hybrid approach, which mixes both semantic and lexical distractors, does not appear to introduce more severe challenges for the models.

Impact of Graph-Based Retrieval. For larger context sizes, using PPR for graph-based reranking in distractor construction provides a significant performance uplift across LLMs and base retrievers. By comparing the solid lines with the dashed lines in Fig. 2, we observe that for nearly every model and retriever, the performance curve paired with PPR is noticeably higher, especially at context sizes of $64K$ and $128K$. This demonstrates that exploiting the relational structure among documents is a powerful method for mitigating distraction. The largest improvement of 44% was observed for Llama-3.1-70B-Instruct with the hybrid retriever, highlighting how prioritizing structurally central documents can mitigate more harmful structurally isolated lexical and semantic distractors.

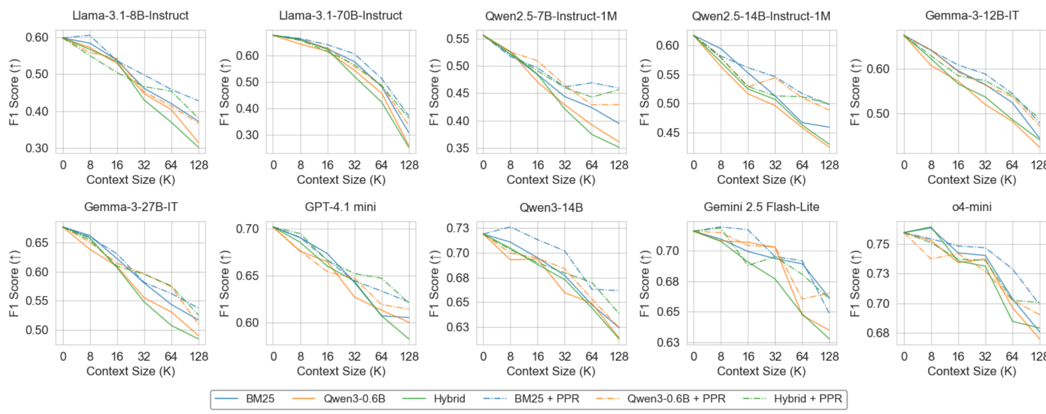

Figure 2: Impact of retriever choice on NIAH performance as context size increases. 0 stands for the case without distractors.

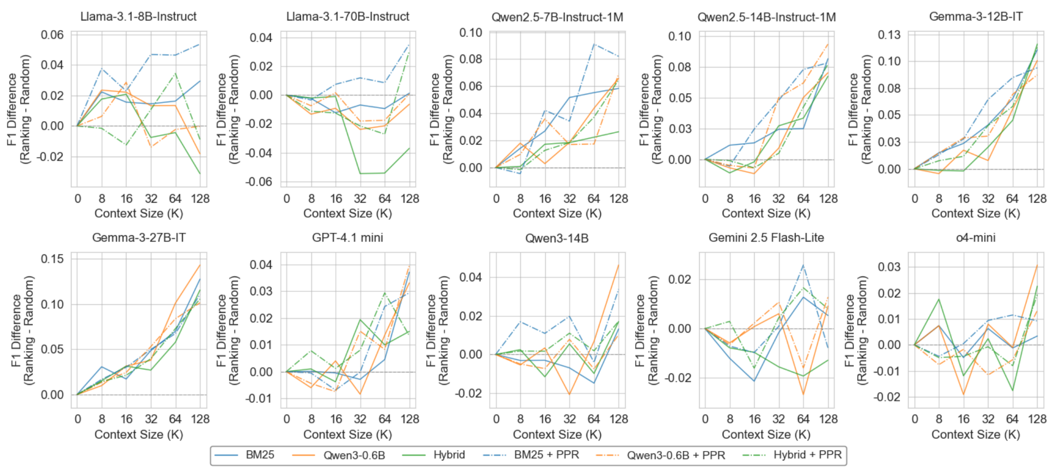

Figure 3: Performance difference in F1 score between using a retriever-ranked ordering and an average of three random permutations.

Retrieval Effectiveness vs NIAH Performance. Jin et al. (2025) suggests that better retrievers introduce harder distractors for shorter-context reasoning and single-hop QA. Our study shows that the interplay between the retriever mechanism and task setting plays a crucial role, where a proper retriever can be simultaneously more effective in retrieval and hard distractor mitigation.

Impact of Haystack Ordering. To isolate the effect of haystack ordering (π), we compare the performance of retriever-ranked ordering against the average of three random permutations. The results in Fig. 3 reveal complex and highly model-dependent patterns. While Gemma-3 and Qwen2.5-1M families derive a significant and growing benefit from retriever-ranked ordering as context size expands, others exhibit a more volatile, retriever-dependent, or even negative response. This finding carries a crucial implication: to faithfully assess a model's practical long-context utility in RAG, evaluations must mirror the canonical, retriever-ranked input. Furthermore, contrasting this setup with random permutations allows us to better understand the positional biases of individual models.

4 Conclusion

We introduce haystack engineering for a principled NIAH benchmark framework. Through our new benchmark, HaystackCraft, we demonstrate that the composition and ordering of the haystack, as determined by heterogeneous retrieval strategies, critically impact model performance.

References

Gordon V. Cormack, Charles L A Clarke, and Stefan Buettcher. Reciprocal rank fusion outperforms condorcet and individual rank learning methods. In *Proceedings of the 32nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 758–759, 2009.

Tri Dao, Daniel Y. Fu, Stefano Ermon, Atri Rudra, and Christopher Ré. FlashAttention: Fast and Memory-Efficient Exact Attention with IO-Awareness. In *Advances in Neural Information Processing Systems*, 2022.

Abhimanyu Dubey et al. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*, 2024.

Kenneth Enevoldsen et al. MMTEB: Massive Multilingual Text Embedding Benchmark. In *International Conference on Learning Representations*, 2025.

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

Cheng-Ping Hsieh, Simeng Sun, Samuel Kriman, Shantanu Acharya, Dima Rekesh, Fei Jia, and Boris Ginsburg. RULER: What's the Real Context Size of Your Long-Context Language Models? In *Conference on Language Modeling*, 2024.

Kalervo Järvelin and Jaana Kekäläinen. IR evaluation methods for retrieving highly relevant documents. In *Proceedings of the 23rd Annual International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 41–48, 2000.

Kalervo Järvelin and Jaana Kekäläinen. Cumulated gain-based evaluation of IR techniques. *ACM Trans. Inf. Syst.*, 20(4):422–446, 2002.

Bowen Jin, Jinsung Yoon, Jiawei Han, and Sercan O Arik. Long-Context LLMs Meet RAG: Overcoming Challenges for Long Inputs in RAG. In *International Conference on Learning Representations*, 2025.

Aishwarya Kamath et al. Gemma 3 Technical Report. *arXiv preprint arXiv:2503.19786*, 2025.

Gregory Kamradt. Needle In A Haystack - Pressure Testing LLMs. `https://github.com/gkamradt/LLMTest_NeedleInAHaystack`. Accessed: Apr. 15, 2025.

Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense Passage Retrieval for Open-Domain Question Answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pp. 6769–6781, 2020.

Yuri Kuratov, Aydar Bulatov, Petr Anokhin, Ivan Rodkin, Dmitry Igorevich Sorokin, Artyom Sorokin, and Mikhail Burtsev. BABILong: Testing the Limits of LLMs with Long Context Reasoning-in-a-Haystack. In *Advances in Neural Information Processing Systems Datasets and Benchmarks Track*, 2024.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Matthew Kelcey, Jacob Devlin, Kenton Lee, Kristina N. Toutanova, Llion Jones, Ming-Wei Chang, Andrew Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. Natural Questions: a Benchmark for Question Answering Research. *Transactions of the Association of Computational Linguistics*, 2019.

Woosuk Kwon, Zhuohan Li, Siyuan Zhuang, Ying Sheng, Lianmin Zheng, Cody Hao Yu, Joseph E. Gonzalez, Hao Zhang, and Ion Stoica. Efficient Memory Management for Large Language Model Serving with PagedAttention. In *Proceedings of the ACM SIGOPS 29th Symposium on Operating Systems Principles*, 2023.

Dohyeon Lee, Seung-won Hwang, Kyungjae Lee, Seungtaek Choi, and Sunghyun Park. On Complementarity Objectives for Hybrid Retrieval. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pp. 13357–13368, 2023.

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*, pp. 9459–9474, 2020.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. Lost in the Middle: How Language Models Use Long Contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173, 2024.

Lingrui Mei, Jiayu Yao, Yuyao Ge, Yiwei Wang, Baolong Bi, Yujun Cai, Jiazhi Liu, Mingyu Li, Zhong-Zhi Li, Duzhen Zhang, Chenlin Zhou, Jiayi Mao, Tianze Xia, Jiafeng Guo, and Shenghua Liu. A Survey of Context Engineering for Large Language Models. *arXiv preprint 2507.13334*, 2025.

Lawrence Page, Sergey Brin, Rajeev Motwani, and Terry Winograd. The PageRank Citation Ranking : Bringing Order to the Web. In *The Web Conference*, 1999.

Bowen Peng, Jeffrey Quesnelle, Honglu Fan, and Enrico Shippole. YaRN: Efficient Context Window Extension of Large Language Models. In *International Conference on Learning Representations*, 2024.

Stephen Robertson and Hugo Zaragoza. The Probabilistic Relevance Framework: BM25 and Beyond. *Found. Trends Inf. Retr.*, 3(4):333–389, 2009.

Stephen E. Robertson, Steve Walker, Susan Jones, Micheline Hancock-Beaulieu, and Mike Gatford. Okapi at TREC-3. In *TREC*, volume 500-225 of *NIST Special Publication*, pp. 109–126, 1994.

Oscar Sainz, Jon Campos, Iker García-Ferrero, Julen Etxaniz, Oier Lopez de Lacalle, and Eneko Agirre. NLP Evaluation in trouble: On the Need to Measure LLM Data Contamination for each Benchmark. In *Findings of the Association for Computational Linguistics: EMNLP 2023*, pp. 10776–10787, 2023.

Jianlin Su, Murtadha Ahmed, Yu Lu, Shengfeng Pan, Wen Bo, and Yunfeng Liu. RoFormer: Enhanced transformer with Rotary Position Embedding. *Neurocomputing*, 568:127063, 2024.

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.

Guangxuan Xiao, Yuandong Tian, Beidi Chen, Song Han, and Mike Lewis. Efficient Streaming Language Models with Attention Sinks. In *International Conference on Learning Representations*, 2024.

An Yang et al. Qwen2.5-1m technical report. *arXiv preprint arXiv:2501.15383*, 2025a.

An Yang et al. Qwen3 Technical Report. *arXiv preprint arXiv:2505.09388*, 2025b.

Xinyu Yang, Tianqi Chen, and Beidi Chen. APE: Faster and Longer Context-Augmented Generation via Adaptive Parallel Encoding. In *International Conference on Learning Representations*, 2025c.

Howard Yen, Tianyu Gao, Minmin Hou, Ke Ding, Daniel Fleischer, Peter Izsak, Moshe Wasserblat, and Danqi Chen. HELMET: How to Evaluate Long-context Models Effectively and Thoroughly. In *International Conference on Learning Representations*, 2025.

Tao Yuan, Xuefei Ning, Dong Zhou, Zhijie Yang, Shiyao Li, Minghui Zhuang, Zheyue Tan, Zhuyu Yao, Dahua Lin, Boxun Li, Guohao Dai, Shengen Yan, and Yu Wang. LV-Eval: A Balanced Long-Context Benchmark with 5 Length Levels Up to 256K. *arXiv preprint arXiv:2402.05136*, 2024.

Yanzhao Zhang, Mingxin Li, Dingkun Long, Xin Zhang, Huan Lin, Baosong Yang, Pengjun Xie, An Yang, Dayiheng Liu, Junyang Lin, Fei Huang, and Jingren Zhou. Qwen3 embedding: Advancing text embedding and reranking through foundation models. *arXiv preprint arXiv:2506.05176*, 2025.

A More Dataset Details

In preparing the Wikipedia hyperlink network, we filter out empty and redirect pages.

Table 1 provides a dataset breakdown over hop count.

Table 1: Question breakdown over hop count.

# hops	%
1	20
2	58
3	15.6
4	6.4

Data Contamination Mitigation. A critical concern in LLM evaluation is data contamination, where exposure to benchmark data during pretraining inflates performance (Sainz et al., 2023). While the models we evaluate have likely been trained on Wikipedia and even the QA datasets, our benchmark's design inherently mitigates this risk. The core task demands in-context reasoning—locating the “needle" within a long context of plausible, retriever-selected distractors—rather than simple fact recall. This challenge is amplified for our multi-hop questions, which require synthesizing information across multiple documents, a process robust to memorization. Furthermore, our use of a recent Wikipedia dump post-dates the training cutoffs of most current LLMs, minimizing data overlap. Crucially, our empirical results confirm this mitigation: all models show substantial performance degradation as context size increases, demonstrating that they are actively reasoning over the provided text, not merely recalling memorized answers.

B Additional Setup Details

B.1 Haystack Construction

The token counts are standardized by the Qwen2.5-1M tokenizer for fair comparison

B.2 LLM Setup

For each LLM, we utilize the recommended inference hyperparameters as specified on its Hugging Face model card. These settings include sampling parameters like temperature, Top-P, Top-K, and Min-P, along with the “thinking budget” for thinking LLMs. All models considered in this work possess native long-context support for at least $128K$ tokens, with the exception of the Qwen3 models. To ensure the Qwen3 models could process a $128K$-token input and generate a $32K$-token output, we extend their context window to $164K$ tokens using YaRN (Peng et al., 2024).

B.3 PPR Setup

We perform a hyperparameter search for PPR per retriever using 10% of the QA samples. For retrieval criteria, we adopt Normalized Discounted Cumulative Gain (NDCG) @ $10K$ (Järvelin & Kekäläinen, 2000, 2002) for ranking ground truth supporting documents among the corpus. Table 2 presents the best hyperparameters for each retriever based on three random seeds.

Table 2: Retriever-specific PPR hyperparameters.

Retriever	# Seed Documents	Damping Factor
BM25	10	0.5
Qwen3-0.6B	5	0.5
Hybrid	5	0.85

C Evaluation for Data Contamination

To quantify data contamination, we evaluate LLM performance under two conditions: 1) without context, to test reliance on parametric knowledge, and 2) with ground-truth supporting documents. We measure F1 scores across an increasing number of question hop to assess how performance varies with reasoning complexity.

Fig. 4 presents the evaluation results.

- **Contamination is evident.** All models achieve non-zero F1 scores without context. This indicates a degree of data contamination.
- **Context is crucial.** Despite contamination, providing ground-truth documents substantially improves the performance of all models.
- **Complexity remains a challenge.** F1 scores generally decrease as the question hop count increases, even when context is provided. This also suggests that evaluation with multi-hop questions suffers less from data contamination.

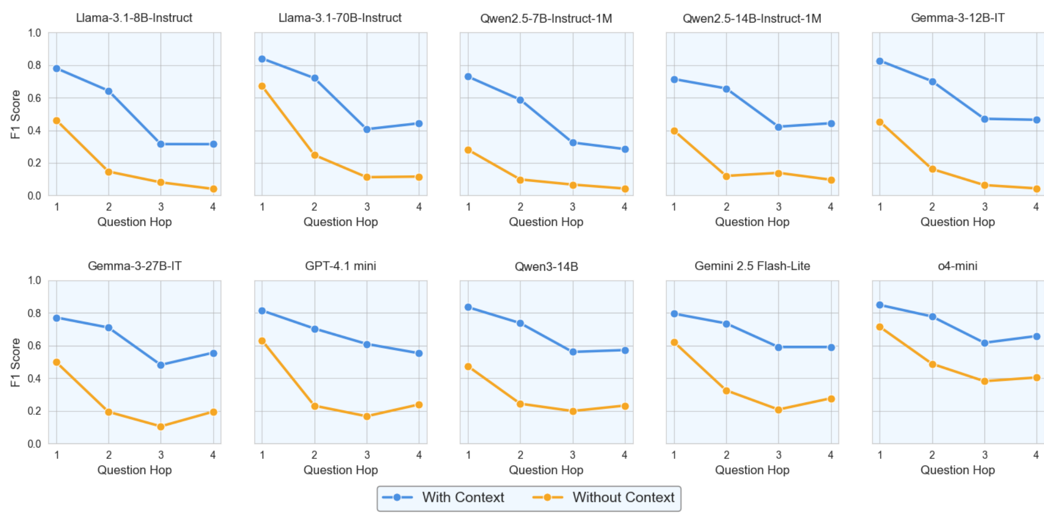

Figure 4: LLM performance with vs without context across question hop.

D Implementation Details

We employ vLLM for LLM inference (Kwon et al., 2023).