FACT: Examining the Effectiveness of Iterative Context Rewriting for Multi-fact Retrieval

Jinlin Wang[1]*, **Suyuchen Wang**[2]*, **Ziwen Xia**[1], **Sirui Hong**[1], **Yun Zhu**[3], **Bang Liu**[2]§, **Chenglin Wu**[1]§

[1]DeepWisdom [2]Université de Montréal & Mila [3]Google

Abstract

Large Language Models (LLMs) are proficient at retrieving single facts from extended contexts, yet they struggle with tasks requiring the simultaneous retrieval of multiple facts, especially during generation. This paper identifies a novel "lost-in-the-middle" phenomenon, where LLMs progressively lose track of critical information throughout the generation process, resulting in incomplete or inaccurate retrieval. To address this challenge, we introduce Find All Crucial Texts (FACT), an iterative retrieval method that refines context through successive rounds of rewriting. This approach enables models to capture essential facts incrementally, which are often overlooked in single-pass retrieval. Experiments demonstrate that FACT substantially enhances multi-fact retrieval performance across various tasks, though improvements are less notable in general-purpose QA scenarios. Our findings shed light on the limitations of LLMs in multi-fact retrieval and underscore the need for more resilient long-context retrieval strategies.

1 Introduction

Large Language Models (LLMs) have demonstrated impressive capabilities in various NLP tasks, particularly in situations where single, salient facts need to be retrieved from a long context (Shi et al., 2023; Izacard and Grave, 2021b; Jiang et al., 2023b; Lin et al., 2023; Jeong et al., 2024). These "needle-in-a-haystack" (gkamradt, 2023) tasks highlight the strength of modern LLMs in isolating critical information (Hsieh et al., 2024; Yoran et al., 2023). However, in tasks requiring the retrieval of multiple facts simultaneously—referred to as *multi-fact retrieval tasks*—the performance of both open-source and proprietary LLMs noticeably degrades (Hsieh et al., 2024; Li et al., 2024a). This is particularly problematic in long-context scenarios (Liu et al., 2023), where models struggle to retain and retrieve multiple pieces of key information, leading to incomplete or erroneous results.

To improve LLMs' performance in multi-fact retrieval tasks, we conducted an analysis of the failure patterns specific to this context. Our findings reveal that the core issue is not identifying relevant information individually but the model's difficulty in focusing on multiple facts as they accumulate. Therefore, in multi-fact retrieval scenarios, as the generation process goes on, the model gradually loses track of the information to be retrieved and tends to retrieve incomplete or incorrect information. This issue, which we term the "lost-in-the-middle" in multi-fact retrieval generation, occurs when multiple critical pieces of information are distributed throughout the context. Conventional retrieval techniques—whether relying on LLM querying or vector-based methods—tend to focus on isolated facts, missing the broader context needed to retrieve all necessary information for complete understanding or reasoning.

Based on this observation, we investigate whether a multi-round retrieval scheme can mitigate performance drops in multi-fact retrieval tasks. Specifically, we introduce **Find All Crucial Texts** (FACT), an iterative approach tailored for multi-fact retrieval. In our method, "context rewriting" leverages previously retrieved information to iteratively refine the context, it's different from "query-rewriting" methods like (Ma et al., 2023; Wang et al., 2023) which obtain additional information by expanding the user's query. . Single-pass retrieval often fails to capture multiple facts, as the model's attention tends to focus primarily on the top-ranked fact. Our iterative process addresses this limitation by progressively removing identified facts from the context, allowing the model to concentrate on additional critical facts in subsequent rounds.

[*] Equal contribution.

[§] Correspondance: Bang Liu (bang.liu@umontreal.ca) and Chenglin Wu (alexanderwu@deepwisdom.ai).

Findings of the Association for Computational Linguistics: NAACL 2025, pages 3382–3392
April 29 - May 4, 2025 ©2025 Association for Computational Linguistics

We empirically demonstrate that FACT significantly outperforms baseline methods in retrieving multiple important facts from long contexts. However, we also show mixed results when applying FACT on general-purpose QA tasks, where we analyze the influence of rewriting rounds, model families, model sizes, and task types on the performance of FACT-like iterative rewriting methods. We conclude that although the multi-round retrieval method drastically benefits retrieval tasks, this performance boost is not universally transferrable to other tasks or models. Our research demonstrates that retrieval tasks themselves are not enough for evaluating long-context scaling methods and calls for better context-building mechanisms, long-context reasoning approaches, and more agentic long-context solutions.

2 The Challenge of Multi-fact Retrieval

In Hsieh et al. (2024), models demonstrate a consistent decline in performance as the number of required retrievals from context increases. This section aims to explore the underlying mechanisms behind this degradation: does the model prematurely terminate its retrieval process, or does it struggle to track and process the necessary information?

We approach this through a mechanistic analysis inspired by Lu et al. (2024). Specifically, we adopt the multi-query needle-in-a-haystack (MQ-NIAH) task from RULER (Hsieh et al., 2024), where the model is presented with a context of key-value pairs and a question containing multiple keys, tasked with retrieving the corresponding values sequentially. To diagnose the model's internal representations, we train a linear probe on each layer of a LLaMA-3 8B Instruct model. The probe maps the intermediate layer representation, $\boldsymbol{x}$, corresponding to an output position of a value, to an output value token y. The probe's accuracy reflects the degree to which the model's internal state retains the necessary information to output the correct value, allowing us to distinguish between cases where the model "knows" the information but fails to output it (high probe accuracy) and cases where the model has entirely lost track of the required information (low probe accuracy). For the details of linear probe training, please refer to Appendix D.

Figure 1 plots the maximum probe accuracy against output position for different query lengths in a 50-query MQ-NIAH setting. The results reveal a clear

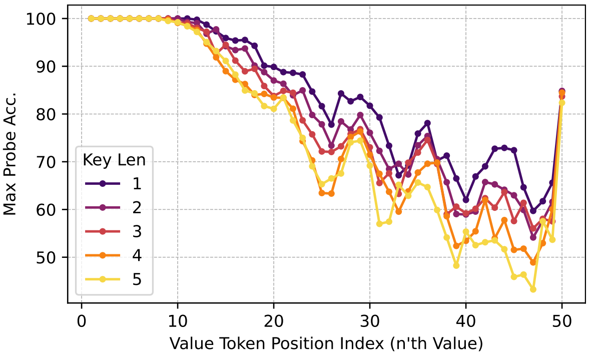

Figure 1: Maximum probing accuracy in a multi-query needle-in-a-haystack (MQ-NIAH) task across all layers of a Llama-3 8B Instruct model. The figure shows a "lost-in-the-middle" phenomenon for the *generation* process in MQ-NIAH.

"lost-in-the-middle" pattern *during generation*: when conducting multi-fact retrieval, the model would progressively lose information of the current retrieval target as the generation continues, until it recovers only at the final few generated tokens.

Please note that this is different from the "lost-in-the-middle *in context*" (Liu et al., 2024), where it loses track of a single retrieval in the middle of the context in single-fact retrieval. The "lost-in-the-middle *during generation*" instead loses track of the current information to be retrieved in the middle of the generation process in multi-fact retrieval.

Notably, the position of the accuracy turning point is largely invariant to the number of key tokens, suggesting that the performance degradation is not due to an overloaded number of key tokens. This pattern implies a fundamental constraint in the model's capacity: **it appears unable to reliably retrieve and track more than a certain number of key pieces of information concurrently from the context**.

3 The FACT Method

As noted above, the facts are basic constituent units within the context. They can be used to provide information in retrieval tasks and to generate answers in QA tasks. The completeness of facts is crucial to retrieval and QA performance. To this end, we introduce an iterative rewriting method called FACT, which significantly enhances fact retrieval performance in common scenarios. This largely addresses the challenges mentioned in Section 2.

3.1 Iterative Rewriting

To solve the problem of incomplete or inaccurate facts, we employ an iterative rewriting approach

Algorithm 1 FACT

```
Require: $Q$: the user query text, $C$: the context of the sample, $n$: number of iterations, Retrieve: the retrieval function, Rewrite: the context rewriting function, Stop: the iteration stop judgment function
Ensure: $F$: the set of final found facts
$F = []$
for $i = 1$ to $n$ do
    $cand\_facts$ = Retrieve($Q$, $C$)
    $C$ = Rewrite($cand\_facts$, $C$)
    $F$.extend($cand\_facts$)
    if Stop($F$, $C$) then
        break
    end if
end for
return $F$
```

for fact retrieval. Specifically, based on the user's query, candidate facts are retrieved through methods such as using LLMs as retrievers or vector-based approaches. These candidate facts are then located inside the context, where they are rewritten by either removing or replaced with other noise data, resulting in a new context. This process is repeated until a stopping criteria is met. In our implementation, the stopping criteria is defined as reaching a predefined number of iterations. The candidate facts found in each iteration are concatenated together to form the final set of facts to be used for generating the final response. The algorithm 1 describes the complete process in detail.

4 Experiments

4.1 Settings

We test the performance of FACT equipped with closed-sourced GPT-4o and GPT-4o-mini (OpenAI, 2024)[§], and open-sourced Llama-3.1 8B Instruct (Dubey et al., 2024). We report the performance on two types of tasks:

- **Retrieval Tasks**, where the model directly retrieves multiple key information in the context. This includes RULER (Hsieh et al., 2024) and Counting Stars (Song et al., 2024a).

- **QA Tasks**, where answering the question requires reasoning about the provided context. This type of task includes: (1) Single-doc QA tasks, including NarrativeQA (Kociský et al., 2018), Qasper (Dasigi et al., 2021), and MultiFieldQA (Bai et al., 2024); (2) Multi-doc QA tasks, including HotpotQA (Yang et al., 2018), 2WikiMQA (Ho et al., 2020), and MuSiQue (Trivedi et al., 2022). The QA tasks adopt the contexts and prompt from LongBench (Bai et al., 2024). Please refer to Appendix B for the statistics of the tasks.

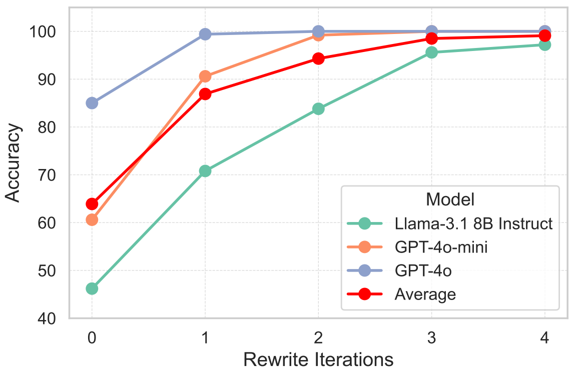

Figure 2: Retrieval Task performances under different numbers of rewriting iterations. The red line denotes the average performance across all tasks.

In the experiments, we compare the results of FACT against a baseline direct retrieval method for each model. This direct retrieval setting returns all the retrieved information or directly answers the question in one shot with the default prompt for each task. We include the prompts we used for the retrieval task and the retrieval step of the QA tasks in Appendix C.

4.2 Retrieval Tasks

We present the results of the retrieval tasks in Table 1. The results demonstrate a significant improvement in retrieval performance when applying our proposed method across both open-source and closed-source models. Across all tasks, the method consistently enhances the models' ability to retrieve as much relevant information as possible from long contexts, outperforming the direct retrieval baselines substantially. This is particularly evident in tasks with longer context lengths, where traditional retrieval methods struggle. Moreover, for better-performing models like GPT-4o and GPT-4o-mini, FACT achieves nearly perfect results.

In Figure 2, we show the comparison of our proposed FACT against baselines under varied iterations with 16K context length in RULER K1V10Q1. Note that the retrieval function, rewriting strategy, and stopping criteria in FACT are closely intertwined, making it challenging to ablate any single component without fundamentally altering the method. Instead, by varying the number of iterations, we gain indirect insight into the

[§] We adopt the `gpt-4o-2024-08-06` and `gpt-4o-mini-2024-07-18` versions in our experiments.

LLM	Method	RULER				Counting Stars		Overall
		K1V10Q1		K5V10Q1		N32		
		4K	16K	4K	16K	4K	16K	
L. 8B	base.	70.0	46.2	80.2	61.0	98.6	80.9	72.8
	FACT	**98.4**	**83.8**	**100.0**	**98.6**	**100.0**	**99.2**	**96.7**
4o-mini	base.	73.4	60.6	81.6	59.6	96.7	72.2	70.0
	FACT	**99.8**	**99.2**	**100.0**	**99.6**	**98.0**	**99.9**	**99.4**
gpt-4o	base.	98.4	85.0	99.8	80.6	99.8	92.7	92.7
	FACT	**100.0**	**100.0**	**100.0**	**99.6**	**100.0**	**100.0**	**99.9**

Table 1: Performance Comparison on Retrieval Tasks. "KxVyQz" denotes adding x needles inside the context and retrieving y values from a single query or z queries. "Ny" denotes retrieving y needles from the context. The best performances of each model on each task are **bolded**. "L. 8B" denotes Llama-3.1 8B Instruct; "4o-mini" denotes GPT-4o-mini.

impact of the iterative rewriting mechanism and its interaction with the stopping criterion. As the iteration increases, the overall scores steadily increase, highlighting the benefit of our iterative rewriting strategy. This is especially true for Llama-3.1 8B Instruct: the model exhibits an increase of nearly 50 percentage points when the iteration count is raised to 3.

4.3 Question Answering Tasks

We present the results of QA Tasks in Figure 3.

Effect of iterative context rewriting on different model families. We include Qwen-2.5 7B (Yang et al., 2024) into discussion in this section. The performance impact of iterative context rewriting varies significantly across model families. GPT-4o and GPT-4o-mini consistently improve as the number of rewriting iterations increases. However, Llama-3.1 and Qwen-2.5 show a noticeable performance decline with iterative retrieval, particularly Llama-3.1, which struggles with retrieved context. This difference likely comes from training differences: GPT-4o may have been specifically trained on retrieval-augmented tasks, while Llama-3.1 and Qwen-2.5 may lack such training, making them more prone to hallucinations or errors.

Iterative rewriting versus one-shot retrieval. Our results show that iterative rewriting outperforms one-shot retrieval, especially for models suited to retrieval-based tasks. Iterative rewriting leads to continuous improvements across iterations, showing the benefits of gradual context refinement. This supports our hypothesis in Section 2, which suggests that repeated enhancement of retrieved context improves model understanding and response quality.

Variability in task-specific performance with iterative rewriting. The impact of iterative rewriting varies significantly across tasks. For the GPT-4o family, we see major gains for datasets like *2wikimqa* and *MuSiQue* but minor declines for *Qasper* and *NarrativeQA*. This is likely due to the different characteristics of each dataset. *2wikimqa* and *MuSiQue* contain dense factual information, which benefits from iterative rewriting by emphasizing key details and reducing noise, thereby improving accuracy. On the other hand, *Qasper* and *NarrativeQA* require nuanced reasoning and complex knowledge, which are beyond mere retrieval. Iterative rewriting in these cases may oversimplify or alter essential information, leading to loss of detail and increased ambiguity. Thus, while factual tasks benefit from FACT, highly structured or narrative tasks may not.

Cause of Failure. Our analysis reveals that open-sourced models fail primarily because they cannot reliably execute atomic retrieval steps in scenarios that require complex reasoning. Note that our proposed method FACT is designed to minimize error accumulation in iterative context rewriting: the output of one round does not directly feed into the next, ruling out the possibility that performance declines arise from compounding errors. Rather, the inability of open-sourced models to retrieve relevant context in each single round consistently hurts their performance, whereas closed-sourced models are able to establish a stronger contextual foundation for context retrieval, likely due to targeted training on retrieval-oriented tasks. This gap in retrieval capability thus stands as the most plausible explanation for the weaker outcomes of open-sourced models, reinforcing the importance of specialized training on fine-grained context retrieval tasks for reliable fact-centric reasoning based on iterative context shortening.

5 Conclusion

This paper explored the challenges faced by Large Language Models (LLMs) in multi-fact retrieval tasks, particularly the "lost-in-the-middle" phenomenon, where models progressively lose track of key facts during generation. To address this, we introduced FACT, an iterative context-rewriting method designed to improve multi-fact retrieval by progressively refining context. Our experiments show that FACT significantly boosts retrieval performance in long-context scenarios, though results

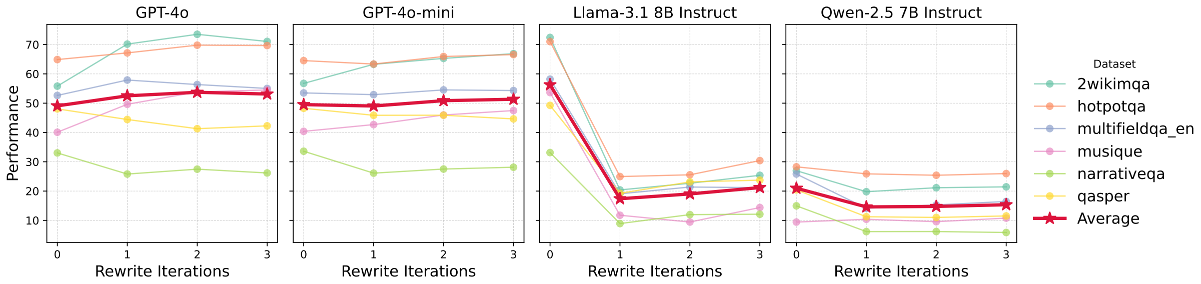

Figure 3: QA Task performances under different numbers of rewriting iterations(retain ten sentences each iterative retrieval)). The red line denotes the average performance across all tasks.

were mixed for general-purpose QA tasks.

These findings underscore the need for robust retrieval mechanisms that go beyond single-pass methods, highlighting the value of iterative refinement in complex retrieval settings. While FACT proves effective for fact-intensive retrieval, its mixed performance on QA tasks suggests further research is needed to adapt iterative methods for broader NLP contexts. Future work should explore dynamic rewriting techniques tailored to task characteristics, balancing context enrichment with the retention of essential information. This could include dataset-aware rewriting strategies that adjust context modification based on task demands, optimizing performance while minimizing trade-offs. Additionally, task-specific training focused on retrieval could enhance the efficacy of iterative context rewriting. Overall, this work lays a foundation for advancing context-building and long-context reasoning methods, pushing the boundaries of multi-fact retrieval capabilities in LLMs.

Limitations

This short paper includes the insights and findings of our experiments to improve LLMs' multi-fact retrieval performance. While the FACT method shows considerable promise in improving multi-fact retrieval performance, there are several aspects that warrant further exploration, which we believe represent opportunities for future work rather than critical shortcomings.

Task-specific Performance Variability. FACT exhibits significant improvements in multi-fact retrieval tasks, but its performance gains in general-purpose QA tasks are more mixed. This variation likely stems from the fundamental differences in task requirements: FACT is particularly well-suited to fact-heavy retrieval tasks, where it refines the context over iterations. However, the iterative approach may not always lead to optimal outcomes in tasks requiring nuanced reasoning or comprehension, such as NarrativeQA or Qasper. Nonetheless, we see this as an opportunity to explore task-adaptive strategies that fine-tune the number of iterations or degree of context rewriting based on specific task characteristics.

Model-specific Behavior. The effectiveness of FACT can differ across model families. Although closed-source models such as GPT-4o benefit significantly from iterative rewriting, some open-source models show smaller gains or sometimes negative gains in retrieval tasks, likely due to differences in training data, regimes and architectures. However, these results highlight the potential to improve the performance of open-source models through targeted training in retrieval-augmented tasks. Addressing this presents an exciting avenue for future research, aiming to make FACT more universally beneficial across various model types.

Computational Considerations. FACT introduces additional computation due to its iterative nature, which could increase latency in certain applications. In practice, this issue can be mitigated by fine-tuning the number of iterations or applying FACT selectively to tasks where its benefits justify the additional cost. Further research on optimizing the efficiency of iterative processes could help minimize this overhead.

Generalization to Broader NLP Tasks. FACT is designed primarily for multi-fact retrieval, and it excels in this task. Its application to more complex reasoning tasks, while promising, has room for improvement. We do not see this as a fundamental limitation of FACT, but rather a natural constraint given its design focus. Adapting FACT to tasks requiring deeper reasoning or synthesis remains an exciting challenge for future research, which could involve integrating more advanced reasoning or agentic procedures into the iterative process.

References

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. 2024. Self-rag: Learning to retrieve, generate, and critique through self-reflection. In *The Twelfth International Conference on Learning Representations, ICLR 2024, Vienna, Austria, May 7-11, 2024*. OpenReview.net.

Yushi Bai, Xin Lv, Jiajie Zhang, Hongchang Lyu, Jiankai Tang, Zhidian Huang, Zhengxiao Du, Xiao Liu, Aohan Zeng, Lei Hou, Yuxiao Dong, Jie Tang, and Juanzi Li. 2024. LongBench: A bilingual, multitask benchmark for long context understanding. In *Proceedings of the 62nd Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 3119–3137, Bangkok, Thailand. Association for Computational Linguistics.

Xin Cheng, Xun Wang, Xingxing Zhang, Tao Ge, Si-Qing Chen, Furu Wei, Huishuai Zhang, and Dongyan Zhao. 2024. xrag: Extreme context compression for retrieval-augmented generation with one token. *arXiv preprint arXiv: 2405.13792*.

Alexis Chevalier, Alexander Wettig, Anirudh Ajith, and Danqi Chen. 2023. Adapting language models to compress contexts. In *Proceedings of the 2023 Conference on Empirical Methods in Natural Language Processing*, pages 3829–3846, Singapore. Association for Computational Linguistics.

Pradeep Dasigi, Kyle Lo, Iz Beltagy, Arman Cohan, Noah A. Smith, and Matt Gardner. 2021. A dataset of information-seeking questions and answers anchored in research papers. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 4599–4610, Online. Association for Computational Linguistics.

Abhimanyu Dubey, Abhinav Jauhri, Abhinav Pandey, Abhishek Kadian, Ahmad Al-Dahle, Aiesha Letman, Akhil Mathur, Alan Schelten, Amy Yang, Angela Fan, Anirudh Goyal, Anthony Hartshorn, Aobo Yang, Archi Mitra, Archie Sravankumar, Artem Korenev, Arthur Hinsvark, Arun Rao, Aston Zhang, Aurelien Rodriguez, Austen Gregerson, Ava Spataru, Baptiste Roziere, Bethany Biron, Binh Tang, Bobbie Chern, Charlotte Caucheteux, Chaya Nayak, Chloe Bi, Chris Marra, Chris McConnell, Christian Keller, Christophe Touret, Chunyang Wu, Corinne Wong, et al. 2024. The llama 3 herd of models. *arXiv preprint arXiv:2407.21783*.

gkamradt. 2023. Needle in a haystack - pressure testing llms. `https://github.com/gkamradt/LLMTest_NeedleInAHaystack`.

Ralph Grishman and Beth M Sundheim. 1996. Message understanding conference-6: A brief history. In *COLING 1996 volume 1: The 16th international conference on computational linguistics*.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. Realm: retrieval-augmented language model pre-training. In *Proceedings of the 37th International Conference on Machine Learning*, ICML'20. JMLR.org.

Xu Han, Hao Zhu, Pengfei Yu, Ziyun Wang, Yuan Yao, Zhiyuan Liu, and Maosong Sun. 2018. Fewrel: A large-scale supervised few-shot relation classification dataset with state-of-the-art evaluation. *arXiv preprint arXiv:1810.10147*.

Marti A Hearst. 1992. Automatic acquisition of hyponyms from large text corpora in proc. In *14th International Conference Computational Linguistics, Nantes France*.

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, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Cheng-Ping Hsieh, Simeng Sun, Samuel Kriman, Shantanu Acharya, Dima Rekesh, Fei Jia, Yang Zhang, and Boris Ginsburg. 2024. Ruler: What's the real context size of your long-context language models? *arXiv preprint arXiv: 2404.06654*.

Gautier Izacard and Edouard Grave. 2021a. Distilling knowledge from reader to retriever for question answering. In *International Conference on Learning Representations*.

Gautier Izacard and Edouard Grave. 2021b. Leveraging passage retrieval with generative models for open domain question answering. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*, pages 874–880, Online. Association for Computational Linguistics.

Soyeong Jeong, Jinheon Baek, Sukmin Cho, Sung Ju Hwang, and Jong C Park. 2024. Adaptive-rag: Learning to adapt retrieval-augmented large language models through question complexity. *arXiv preprint arXiv:2403.14403*.

Huiqiang Jiang, Qianhui Wu, Xufang Luo, Dongsheng Li, Chin-Yew Lin, Yuqing Yang, and Lili Qiu. 2023a. Longllmlingua: Accelerating and enhancing llms in long context scenarios via prompt compression. *Preprint*, arXiv:2310.06839.

Zhengbao Jiang, Antonios Anastasopoulos, Jun Araki, Haibo Ding, and Graham Neubig. 2020. X-factr: Multilingual factual knowledge retrieval from pretrained language models. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5943–5959.

Zhengbao Jiang, Frank F. Xu, Luyu Gao, Zhiqing Sun, Qian Liu, Jane Dwivedi-Yu, Yiming Yang, Jamie Callan, and Graham Neubig. 2023b. Active retrieval augmented generation. *arXiv preprint arXiv:2305.06983*.

Jacob Devlin Ming-Wei Chang Kenton and Lee Kristina Toutanova. 2019. Bert: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of naacL-HLT*, volume 1, page 2. Minneapolis, Minnesota.

Tushar Khot, Harsh Trivedi, Matthew Finlayson, Yao Fu, Kyle Richardson, Peter Clark, and Ashish Sabharwal. 2023. Decomposed prompting: A modular approach for solving complex tasks. In *The Eleventh International Conference on Learning Representations*.

Tomás Kociský, Jonathan Schwarz, Phil Blunsom, Chris Dyer, Karl Moritz Hermann, Gábor Melis, and Edward Grefenstette. 2018. The narrativeqa reading comprehension challenge. *Trans. Assoc. Comput. Linguistics*, 6:317–328.

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. 2020. Retrieval-Augmented Generation for knowledge-intensive NLP tasks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474.

Mo Li, Songyang Zhang, Yunxin Liu, and Kai Chen. 2024a. Needlebench: Can llms do retrieval and reasoning in 1 million context window? *arXiv preprint arXiv:2407.11963*.

Xiaonan Li, Changtai Zhu, Linyang Li, Zhangyue Yin, Tianxiang Sun, and Xipeng Qiu. 2024b. LLatrieval: LLM-verified retrieval for verifiable generation. In *Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers)*, pages 5453–5471, Mexico City, Mexico. Association for Computational Linguistics.

Xi Victoria Lin, Xilun Chen, Mingda Chen, Weijia Shi, Maria Lomeli, Rich James, Pedro Rodriguez, Jacob Kahn, Gergely Szilvasy, Mike Lewis, et al. 2023. Ra-dit: Retrieval-augmented dual instruction tuning. *arXiv preprint arXiv:2310.01352*.

Nelson F. Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2023. Lost in the middle: How language models use long contexts. *Preprint*, arXiv:2307.03172.

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.

Taiming Lu, Muhan Gao, Kuai Yu, Adam Byerly, and Daniel Khashabi. 2024. Insights into llm long-context failures: When transformers know but don't tell. *arXiv preprint arXiv: 2406.14673*.

Kun Luo, Zheng Liu, Shitao Xiao, and Kang Liu. 2024. Bge landark embedding: A chunking-free embedding method for retrieval augmented long-context large language models. *Preprint*, arXiv:2402.11573.

Xinbei Ma, Yeyun Gong, Pengcheng He, Hai Zhao, and Nan Duan. 2023. Query rewriting for retrieval-augmented large language models. *arXiv preprint arXiv:2305.14283*.

Mike Mintz, Steven Bills, Rion Snow, and Dan Jurafsky. 2009. Distant supervision for relation extraction without labeled data. In *Proceedings of the Joint Conference of the 47th Annual Meeting of the ACL and the 4th International Joint Conference on Natural Language Processing of the AFNLP*, pages 1003–1011.

Jesse Mu, Xiang Li, and Noah Goodman. 2024. Learning to compress prompts with gist tokens. *Advances in Neural Information Processing Systems*, 36.

OpenAI. 2024. Gpt-4o system card.

Md Rizwan Parvez. 2024. Evidence to generate (e2g): A single-agent two-step prompting for context grounded and retrieval augmented reasoning. *arXiv preprint arXiv:2401.05787*.

Hongjin Qian, Zheng Liu, Kelong Mao, Yujia Zhou, and Zhicheng Dou. 2024. Grounding language model with chunking-free in-context retrieval. *Preprint*, arXiv:2402.09760.

Weijia Shi, Sewon Min, Michihiro Yasunaga, Minjoon Seo, Rich James, Mike Lewis, Luke Zettlemoyer, and Wen tau Yih. 2023. Replug: Retrieval-augmented black-box language models. *arXiv preprint arXiv:2301.12652*.

Noah Shinn, Federico Cassano, Ashwin Gopinath, Karthik Narasimhan, and Shunyu Yao. 2023. Reflexion: language agents with verbal reinforcement learning. In *Advances in Neural Information Processing Systems 36: Annual Conference on Neural Information Processing Systems 2023, NeurIPS 2023, New Orleans, LA, USA, December 10 - 16, 2023*.

Mingyang Song, Mao Zheng, and Xuan Luo. 2024a. Counting-stars: A multi-evidence, position-aware, and scalable benchmark for evaluating long-context large language models. *arXiv preprint arXiv: 2403.11802*.

Woomin Song, Seunghyuk Oh, Sangwoo Mo, Jaehyung Kim, Sukmin Yun, Jung-Woo Ha, and Jinwoo Shin. 2024b. Hierarchical context merging: Better long context understanding for pre-trained llms. *arXiv preprint arXiv:2404.10308*.

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.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2023. Interleaving retrieval with chain-of-thought reasoning for knowledge-intensive multi-step questions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 10014–10037, Toronto, Canada. Association for Computational Linguistics.

Fei Wang, Xingchen Wan, Ruoxi Sun, Jiefeng Chen, and Sercan Ö Arık. 2024. Astute rag: Overcoming imperfect retrieval augmentation and knowledge conflicts for large language models. *arXiv preprint arXiv:2410.07176*.

Liang Wang, Nan Yang, and Furu Wei. 2023. Query2doc: Query expansion with large language models. *arXiv preprint arXiv:2303.07678*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Brian Ichter, Fei Xia, Ed H. Chi, Quoc V. Le, and Denny Zhou. 2022. Chain-of-thought prompting elicits reasoning in large language models. In *Advances in Neural Information Processing Systems 35: Annual Conference on Neural Information Processing Systems 2022, NeurIPS 2022, New Orleans, LA, USA, November 28 - December 9, 2022*.

Chengmin Wu, Enrui Hu, Ke Zhan, Lan Luo, Xinyu Zhang, Hao Jiang, Qirui Wang, Zhao Cao, Fan Yu, and Lei Chen. 2022. Triple-fact retriever: An explainable reasoning retrieval model for multi-hop qa problem. In *2022 IEEE 38th International Conference on Data Engineering (ICDE)*, pages 1206–1218. IEEE.

Peng Xu, Wei Ping, Xianchao Wu, Lawrence McAfee, Chen Zhu, Zihan Liu, Sandeep Subramanian, Evelina Bakhturina, Mohammad Shoeybi, and Bryan Catanzaro. 2023. Retrieval meets long context large language models. In *The Twelfth International Conference on Learning Representations*.

Shi-Qi Yan, Jia-Chen Gu, Yun Zhu, and Zhen-Hua Ling. 2024. Corrective retrieval augmented generation. *arXiv preprint arXiv: 2401.15884*.

An Yang, Baosong Yang, Binyuan Hui, Bo Zheng, Bowen Yu, Chang Zhou, Chengpeng Li, Chengyuan Li, Dayiheng Liu, Fei Huang, Guanting Dong, Haoran Wei, Huan Lin, Jialong Tang, Jialin Wang, Jian Yang, Jianhong Tu, Jianwei Zhang, Jianxin Ma, Jianxin Yang, Jin Xu, Jingren Zhou, Jinze Bai, Jinzheng He, Junyang Lin, Kai Dang, Keming Lu, Keqin Chen, Kexin Yang, Mei Li, Mingfeng Xue, Na Ni, Pei Zhang, Peng Wang, Ru Peng, Rui Men, Ruize Gao, Runji Lin, Shijie Wang, Shuai Bai, Sinan Tan, Tianhang Zhu, Tianhao Li, Tianyu Liu, Wenbin Ge, Xiaodong Deng, Xiaohuan Zhou, Xingzhang Ren, Xinyu Zhang, Xipin Wei, Xuancheng Ren, Xuejing Liu, Yang Fan, Yang Yao, Yichang Zhang, Yu Wan, Yunfei Chu, Yuqiong Liu, Zeyu Cui, Zhenru Zhang, Zhifang Guo, and Zhihao Fan. 2024. Qwen2 technical report. *arXiv preprint arXiv: 2407.10671*.

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, Brussels, Belgium. Association for Computational Linguistics.

Yuan Yao, Deming Ye, Peng Li, Xu Han, Yankai Lin, Zhenghao Liu, Zhiyuan Liu, Lixin Huang, Jie Zhou, and Maosong Sun. 2019. Docred: A large-scale document-level relation extraction dataset. *arXiv preprint arXiv:1906.06127*.

Ori Yoran, Tomer Wolfson, Ori Ram, and Jonathan Berant. 2023. Making retrieval-augmented language models robust to irrelevant context. *arXiv preprint arXiv:2310.01558*.

Zhenrui Yue, Honglei Zhuang, Aijun Bai, Kai Hui, Rolf Jagerman, Hansi Zeng, Zhen Qin, Dong Wang, Xuanhui Wang, and Michael Bendersky. 2024. Inference scaling for long-context retrieval augmented generation. *arXiv preprint arXiv:2410.04343*.

Tianjun Zhang, Shishir G Patil, Naman Jain, Sheng Shen, Matei Zaharia, Ion Stoica, and Joseph E Gonzalez. 2024a. Raft: Adapting language model to domain specific rag. *arXiv preprint arXiv:2403.10131*.

Yusen Zhang, Ruoxi Sun, Yanfei Chen, Tomas Pfister, Rui Zhang, and Sercan Ö. Arik. 2024b. Chain of agents: Large language models collaborating on long-context tasks. *arXiv preprint arXiv: 2406.02818*.

Yun Zhu, Jia-Chen Gu, Caitlin Sikora, Ho Ko, Yinxiao Liu, Chu-Cheng Lin, Lei Shu, Liangchen Luo, Lei Meng, Bang Liu, et al. 2024. Accelerating inference of retrieval-augmented generation via sparse context selection. *arXiv preprint arXiv:2405.16178*.

A Related Work

A.1 Information Extraction

Information Extraction (IE) has evolved significantly from rule-based approaches (Grishman and Sundheim, 1996) to modern neural methods. Early work focused on pattern matching and handcrafted features (Hearst, 1992), while recent advances leverage pre-trained language models like BERT (Kenton and Toutanova, 2019) for better performance. Distant supervision (Mintz et al., 2009) has enabled large-scale IE by automatically generating training data. Current research emphasizes few-shot learning (Han et al., 2018) and document-level IE (Yao et al., 2019) to capture complex relations across longer contexts.

A.2 Retrieval-Augmented Generation (RAG)

RAG has been shown to be effective in improving LLM by integrating relevant information retrieved from external sources (Lewis et al., 2020; Guu et al., 2020). This method, notably less computationally costly, has shown success in various tasks such as language modeling and question answering (Shi et al., 2023; Izacard and Grave, 2021b; Jiang et al., 2023b; Lin et al., 2023; Jeong et al., 2024). Despite their effectiveness, traditional RAG methods often suffer from the loss of semantic coherence due to context chunking (Izacard and Grave, 2021a; Xu et al., 2023). Recent approaches aim to mitigate these issues by improving retrieval quality through correction, critique, or verification mechanism (Yan et al., 2024; Asai et al., 2024; Li et al., 2024b; Yoran et al., 2023; Zhang et al., 2024a; Zhu et al., 2024).

The integration of chunking-free methods has further refined the process, enabling more coherent surrogate contexts (Qian et al., 2024; Luo et al., 2024). Techniques such as sequential processing and decision-making frameworks add an additional layer of sophistication, allowing dynamic context handling and improving overall performance in long-context tasks (Wei et al., 2022; Shinn et al., 2023).

A.3 Context Rewriting Methods

Context compression techniques play a crucial role in managing long-context input by reducing the cost of inference while preserving essential information. Methods such as gisting and selective context evaluation have shown the potential to maintain core information through compression (Mu et al., 2024; Jiang et al., 2023a; Wang et al., 2024). xRAG and AutoCompressors further enhance these capabilities by embedding and transforming segments into more compact forms (Cheng et al., 2024; Chevalier et al., 2023), while (Song et al., 2024b) propose to use a divide-and-conquer strategy to hierarchically merge context embeddings across transformer layer. Other works such as (Ma et al., 2023; Wang et al., 2023; Jiang et al., 2020; Wu et al., 2022) focuses on improving LLM's ability to acquire more knowledge from the perspective of query optimization. Moreover, iterative frameworks (Zhang et al., 2024b; Trivedi et al., 2023; Yue et al., 2024; Parvez, 2024) retrieves relevant information iteratively throughout the reasoning steps, and facilitate information aggregation and reasoning over extended contexts. These methods aim to retain valuable information by linking and synthesizing pivotal segments, thus improving the efficiency and effectiveness of long-context LLMs in multi-hop reasoning tasks (Khot et al., 2023; Trivedi et al., 2023).

B Dataset Statistics

Dataset	NQA	Qasper	MFQA	HotpotQA	2Wiki	MuSiQue
#Samples	200	200	150	200	200	200
Avg Length	18,409	3,619	4,559	9,151	4,887	11,214
Metric	F1	F1	F1	F1	F1	F1

Table 2: Statistics of the QA datasets.

In this section, we provide the dataset statistics for the QA datasets we used in Section 4 in Table 2. These datasets are derived from LongBench (Bai et al., 2024). For the retrieval datasets, please refer to the configurations specified in Table 1.

C Prompts

For the retrieval tasks and QA tasks, we use the official prompt provied by RULER (Hsieh et al., 2024) or Counting Stars (Song et al., 2024a), and LongBench (Bai et al., 2024), respectively. We provide the prompt template we used for FACT's retrieval step for all the evaluated tasks below.

Prompt used for FACT's retrieval step for the RULER Retrieval tasks

Some special magic numbers are hidden within the following text. Make sure to memorize it. I will quiz you about the numbers afterwards.

{context}

What are all the special magic numbers for {query} mentioned in the provided text? The special magic numbers for {query} mentioned in the provided text are

Prompt used for FACT's retrieval step for the Counting-Stars tasks

{context1}
The little penguin counted {number1} ∗
{context2}
The little penguin counted {number2} ∗

On this moonlit and misty night, the little penguin is looking up at the sky and concentrating on counting ∗. Please help the little penguin collect the number of ∗, for example: {"little_penguin": [x, x, x,...]}. The summation is not required, and the numbers in [x, x, x,...] represent the counted number of ∗by the little penguin. Only output the results in JSON format without any explanation.

Prompt used for FACT's retrieval step for the QA tasks

Please retrieve all the sentences in the given documents that are important and relevant to answer the question.

Question: {question}

The following are given documents.

{context}

Please retrieve the sentences from the given documents that are relevant to answer the question. Do not repeat your generation. The question is highlighted again at below.

Question: {question}

Retrieved sentences:

(For each retrieved sentence, please start from the bullet symbol "-", if no results, just return a single "-")

D Linear Probe Training Details

This section describes the training process for the linear probes used in Section 2, specifically for the MQ-NIAH task. This linear probe is a multi-fact retrieval extension of the one proposed by Lu et al. (2024).

For the MQ-NIAH task introduced in Section 2, the model receives n_q queries and must retrieve corresponding values from n_k key-value pairs in the prompt, where each value consists of a single token. We define $\mathbb{V}$ as the set of all possible single-token values. Given a prompt, we define the index of the token corresponding to the i-th output value as $t_i \in \mathbb{R}$, and the value token itself as $v_i \in \mathbb{V}$.

Assume the LLM consists of L layers. For each Transformer layer, we randomly initialize a linear classifier $\boldsymbol{C} \in \mathbb{R}^{d \times v}$, where d is the hidden dimension of the LLM, and $v = |\mathbb{V}|$ is the number of possible values. Given the output from the l-th layer, denoted as $H_l \in \mathbb{R}^{L \times d}$ (with L representing the sequence length), the linear classifier $\boldsymbol{C}_l$ predicts the value v_i using the hidden state $H_{l,[t_i,:]}$ for each $i \in \{1, \cdots, n_q\}$.

We collect training data and conduct inference using a specifically designed prompt. During training, we concatenate the ground-truth values to the prompt and record v_i and $H_{l,[t_i,:]}$ for all layers $l \in \{1, \cdots, L\}$ and queries $i \in \{1, \cdots, n_q\}$ in a single forward pass. The training prompt is structured as follows:

Prompt used for MQ-NIAH task in Section 2

Extract the value corresponding to the specified key in the JSON object below.

{"|"_separated_keys}

JSON data: {json_formatted_key_value_pairs}

Keys: {"|"_separated_keys}

Corresponding Value:

In our experiments, we use Llama-3 8B Instruct (Dubey et al., 2024) as the LLM, and we assign $n_q = 50$, $n_k = 200$. The linear classifiers are trained with the hyperparameters specified in Table 3. All experiments are done with either a single NVIDIA RTX3090 24G or a single NVIDIA A100 40G.

Key	#Samples	Epoch	Learning Rate
Value	2000	150	0.005

Table 3: Hyperparameters of Linear Probe Training.

E Cost Analysis

In this section, we present additional experiments measuring the total token throughput in various retrieval scenarios. The results summarized in Table 4 compare the computational efficiency of the Base and FACT methods across different LLM configurations and benchmarks (RULER K1V10Q1, RULER K5V10Q1, and Counting Stars N32). Although the iterative procedure of FACT introduces some computational overhead, the overall token throughput remains within acceptable ranges for practical applications compared to the performance gains it brings. These findings provide further insight into the efficiency trade-offs associated with FACT.

		RULER				**Counting Stars**	
LLM	Method	K1V10Q1	Ratio	K5V10Q1	Ratio	Counting Stars	Ratio
4o-mini	Base	3853.2	/	3674.02	/	3412.18	/
4o-mini	FACT	10446.68	2.71	11152.86	3.04	8596.38	2.52
4o	Base	3860.52	/	3676.81	/	3414.64	/
4o	FACT	7982.52	2.07	7252.69	1.97	9084.18	2.66

Table 4: Average Token Throughput in Retrieval Scenarios for Base and FACT when applying FACT for three iterations. Ratio: the token throughput ratio of FACT to the baseline method.