A Scalable Two-Phase Method for Low-Overhead Retrieval-Augmented Language Models

Kadhim Hayawi
AICEO
Arjaan Office Tower 302, AL Sufouh Complex, Tecom
Dubai, UAE
kadhim.hayawi@ceoai.ai

Sakib Shahriar
AICEO
Arjaan Office Tower 302, AL Sufouh Complex, Tecom
Dubai, UAE
sakib.shahriar@ceoai.ai

***Abstract*—Recent advances in Retrieval-Augmented Generation (RAG) have allowed large language models (LLMs) to consult external knowledge bases to overcome their parameterized knowledge limitations. However, many RAG-based systems depend on iterative procedures like multi-step question decomposition, which can become unnecessarily time-consuming when questions are straightforward. To address this issue, we propose STAR (Self-Knowledge and Targeted Augmented Retrieval), a simplified two-branch retrieval-augmented question-answering framework composed of two main components: a Self-Knowledge Branch and an optimized Passage Retrieval Branch. In our design, the system first checks whether the question can be answered using the LLM's existing knowledge. If not, it retrieves passages and then ranks smaller segments of those passages based on their relevance. This approach reduces the query's reliance on extraneous content, lowers computational costs, and offers a more direct route to accurate solutions. Empirical results on benchmarks such as Natural Questions and TriviaQA show that our simplified approach matches or exceeds the accuracy of multi-module counterparts while operating at a fraction of the complexity.**

Keywords—Retrieval-augmented generation, large language models, contextual learning, question-answering, few-shot learning

I. INTRODUCTION

Advances in large language models (LLMs) have now reached a point where they can handle a broad range of natural language processing tasks, including straightforward question-answering [1] and more complex multi-step reasoning ([2], [3]). While LLMs have demonstrated substantial capacity in reasoning [4], they inherently rely on parameterized knowledge, which can become outdated or insufficient in addressing some questions. Retrieval-augmented generation (RAG) techniques have emerged to mitigate this limitation by allowing models to look up external knowledge sources rather than relying exclusively on what is encoded in their parameters [5].

In many existing retrieval-augmented frameworks, the process involves multiple components. For instance, it could include a module that assesses whether the model can directly answer a query (self-knowledge), another module that evaluates the relevance of retrieved passages, and another that decomposes complex queries into smaller sub-questions [6]. While this iterative structure can benefit scenarios requiring multi-step reasoning, it may also introduce overhead. For less complex questions, unnecessary decomposition and multiple retrieval rounds can burden the system with extra computation and potential redundancy, hindering scalability and efficiency. The tension between comprehensive query handling and efficient performance remains a core issue. Although iterative decomposition is beneficial for some questions requiring multi-hop reasoning, it may be unnecessary and even counterproductive for straightforward queries. This discrepancy motivates a more streamlined approach that recognizes when an LLM already possesses the knowledge to answer a question and when it needs external information without automatically resorting to iterative decomposition.

Our goal is to develop a retrieval-augmented framework that minimizes unnecessary overhead by avoiding multi-step decomposition when it does not add value. We also aim to enhance retrieval focus, ensuring only the most relevant external information is used to augment the LLM's responses. Further, maintaining or improving the accuracy is also an objective to retain the benefits of augmentation while eliminating excessive complexity. To this end, we propose a two-branch **s**elf-knowledge and **t**argeted **a**ugmented **r**etrieval (STAR) question-answering (QA) architecture. In this architecture, a self-knowledge branch evaluates whether the model can solve the query with its internal knowledge alone. If self-knowledge is insufficient, a passage retrieval branch activates to retrieve concise evidence chunks based on their relevance to the query. By segmenting and ranking the retrieved text, the system focuses on only the most pertinent information, reducing computational cost and token usage. This structure avoids repetitive decomposition and ensures a more direct route to accurate responses. The key contributions of this work include:

1. A streamlined retrieval paradigm that eliminates unnecessary iterative decomposition, thus lowering the computational load for simpler questions.

979-8-3315-3559-9/25/$31.00 ©2025 IEEE

2. An optimized passage retrieval strategy that segments and scores the retrieved evidence to maximize relevance and preserve the token budget.
3. Empirical validation shows that this two-branch approach can match or exceed the performance of iterative methods while offering improved scalability and runtime efficiency.

II. Related Works

LLMs often rely on knowledge encoded in their parameters, which can be incomplete or outdated over time [7]. Retrieval-augmented approaches address this challenge by allowing models to query external information, typically from large corpora [8]. These methods have demonstrated improvements in factual accuracy and the ability to incorporate new/emerging knowledge ([9], [10]). However, their effectiveness heavily depends on retrieving highly relevant passages; irrelevant or noisy content can degrade performance [11]. Several works have proposed self-knowledge and dynamic retrieval methods to improve RAG limitations. This includes investigating when a model should rely on its own parameterized knowledge versus when it should search for external evidence [12], [13]. Approaches like Self-RAG [12] introduce specialized tokens and reflection mechanisms for the model to decide if retrieval is necessary. Meanwhile, SKR employs a small classifier to detect whether the model "knows" the answer [13]. These strategies aim to avoid extra retrieval for straightforward questions to streamline computation. Other approaches in the literature have explored iterative decomposition and multi-hop queries. For instance, many RAG systems incorporate decomposition modules to break complex questions into smaller subproblems [14], [15]. Likewise, RA-ISF leverages three submodules, a Self-Knowledge Module, a Passage Relevance Module, and a Question Decomposition Module, to reduce hallucinations and improve answer quality when facing multi-turn or multi-hop queries [6]. Though efficient for complex tasks, iterative decomposition can lead to extra retrieval calls, thereby introducing additional latency and token usage, especially when simpler queries do not benefit from decomposition [11]. In contrast, our method draws on the insights of iterative frameworks but strategically omits the question decomposition stage for cases where it provides limited benefit. By focusing on two key branches (self-knowledge assessment and targeted passage retrieval), we aim to mitigate redundant computation while maintaining a notable question-answering accuracy. This balanced design can be viewed as a hybrid that selectively retains the advantages of retrieval augmentation and self-awareness without incurring the overhead of complex multi-step decomposition pipelines.

III. Proposed Method

Our proposed method streamlines retrieval-augmented question answering by incorporating two primary branches, Self-Knowledge and Passage Retrieval while omitting the iterative Question Decomposition module found in some recent frameworks. Figure 1 summarizes the methodology employed in this study. Next, we provide an overview of the system's workflow, followed by details of each branch and a discussion of how our approach handles text segmentation and relevance scoring.

Identify applicable funding agency here. If none, delete this text box.

A. Approach Overview

The following four steps illustrate the approach we implement:

1. User Query Input. A user question q arrives at the system.
2. Self-Knowledge Branch. The method checks whether the LLM on its own can sufficiently answer the question based on its inherent (parameterized) knowledge. This step prevents unnecessary retrieval when the LLM's internal knowledge is already adequate.
3. Passage Retrieval Branch. If the model indicates insufficient knowledge, the query proceeds to a retrieval module, which fetches and filters potentially relevant external passages. The top passages are segmented into smaller chunks and reordered by relevance. These high-relevance segments are then concatenated with the original query and passed to the LLM for a final answer.
4. Answer Generation. The system produces a concise answer by focusing on the best-matched evidence from the retrieval step.

This approach reduces computational overhead by avoiding multi-step decomposition in situations where iterative question breakdown offers limited benefits. Instead, it relies on a direct (single) retrieval stage and a refined chunk-scoring mechanism.

B. Knowledge Branch and Retrieval

The self-knowledge branch determines whether the question can be answered immediately without external retrieval. First, we perform an internal assessment. This is where a small classification model (or a few-shot prompt) decides if the LLM "knows" the answer from its parameterized knowledge alone. If it does, the system outputs that answer directly. To ensure minimal overhead, we include a further step. Compared to multi-step decomposition, this step is computationally cheap: it simply involves a pass through a shallow network or a single inference call that yields a binary decision ("know" vs. "not know"). This branch effectively screens out queries for which external references are unnecessary to reduce time and resource consumption.

Next, queries determined to be "not known" move ahead to the passage retrieval branch, which refines the retrieval and passage selection steps as follows:

1. Retrieval: We rely on a standard dense-retriever (a Contriever-MS-Marco model) [16] to pull the top-k documents from a Wikipedia-like corpus.
2. Segmentation and Scoring: Each retrieved document is split into smaller chunks (e.g., 100-word blocks). A relevance model then computes similarity scores between each chunk and the user query. This step ensures that only the highest-scoring and most pertinent segments remain.
3. Chunk Ranking and Filtering: The top-scoring segments are concatenated in order of descending

relevance, discarding any irrelevant parts that could introduce noise into the model's final inference step.

4. LLM Integration: The LLM then sees a prompt combining the user query with the pruned segments, enabling it to ground its answer in external evidence.

Through this design, the system leverages only a single retrieval pass, unlike iterative methods that might re-retrieve multiple times or break the query down further. Therefore, the proposed approach keeps the input compact and maximally informative by focusing on chunk-level relevance.

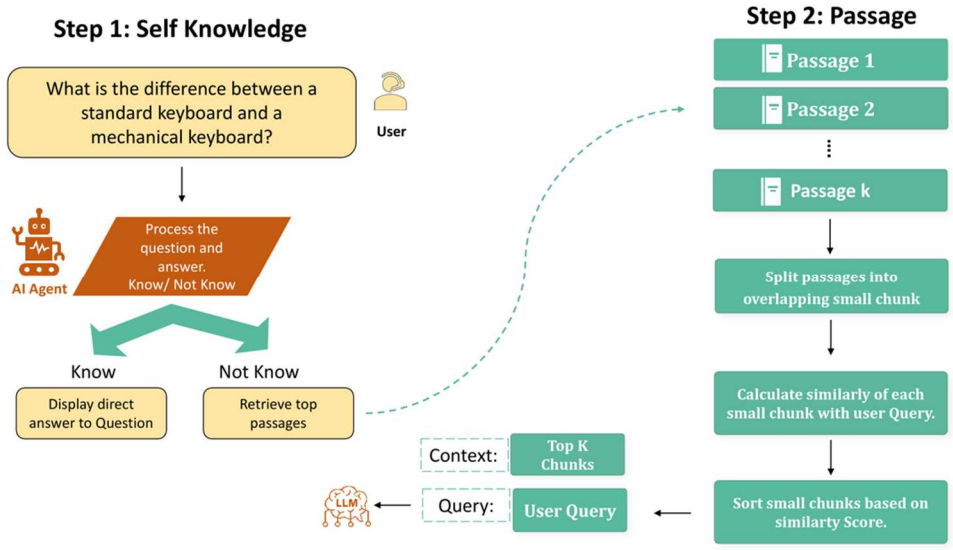

Fig. 1. Summary of the Proposed Self-knowledge with Targeted Augmented Retrieval (STAR) Methodology

C. *Implementation Details*

Figure 2 illustrates how a user's query moves through our two-branch architecture to generate an answer. First, the query arrives at an AI Agent that checks whether the LLM is "sure" about its internal knowledge. If the agent deems it uncertain, the query transitions to a *Not Sure* branch, prompting a retrieval step. The system then consults a vector store, fetching the top-most relevant documents. Next, a final module refines this information by splitting the retrieved documents into smaller chunks and computing their cosine similarity relative to the query. Based on those scores, the system ranks the chunks and selects the top few subparts (such as three highly relevant segments) to form the final context appended to the original query. Finally, the LLM uses this augmented context to produce a more accurate and evidence-based answer. This structured flow avoids unnecessary multi-step decomposition, ensuring the model retrieves external data only when its knowledge is insufficient.

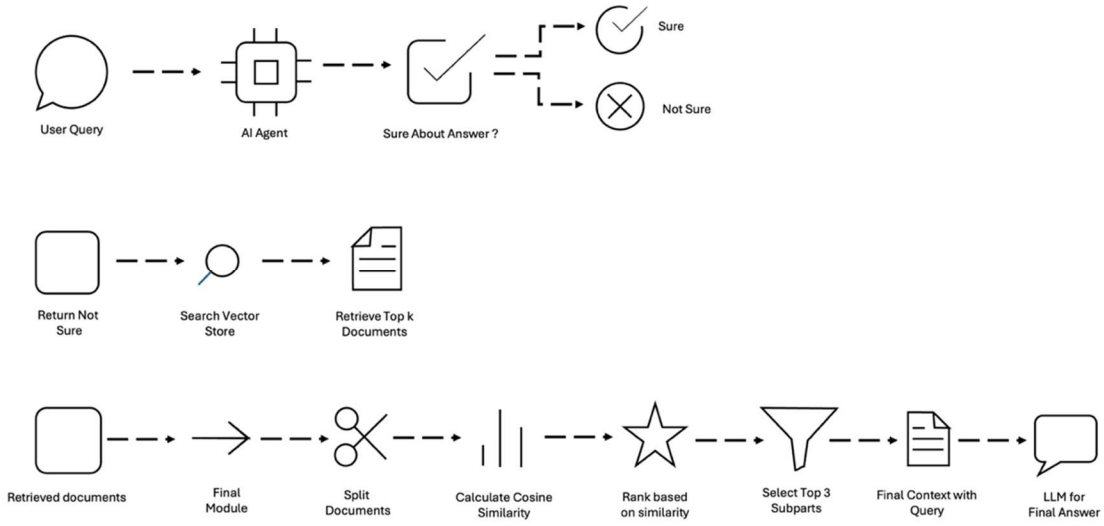

Fig. 2. Summary of the Proposed Self-knowledge with Targeted Augmented Retrieval (STAR) Methodology

D. Experiment Details

We adopt a relatively small chunk size (100 tokens) to prevent less relevant text from diluting the final prompt. A threshold or top-*M* chunk limit is applied (we consider the top 3 chunks) to prioritize the most pertinent segments, thus reducing the token usage for LLMs having input length constraints. The self-knowledge branch can be operationalized in two ways. First, a small, fine-tuned classification model (like a 7B variant of Llama) that outputs a "know" vs. "not know" decision. Second, a lightweight zero-shot or few-shot prompt is appended to the user query, prompting the LLM to self-assess its confidence. Although our approach does not explicitly decompose questions into sub-parts, it can still handle complex queries as long as relevant passages are retrieved. Additionally, given the smaller prompt size, the LLM retains sufficient context to perform advanced reasoning without the overhead of question decomposition. The detailed code and implementation of our approach are made available on GitHub for researchers (https://github.com/sakibsh/STAR).

IV. Results

We evaluate our two-branch retrieval-augmented QA system on four standard benchmarks: Natural Questions (NQ) [17], TriviaQA [18], HotpotQA [19], and StrategyQA [20]. We use two large language model evaluators, Llama2-13B and GPT-3.5, for assessment. We compare against several baselines, including methods without retrieval (Vanilla LM, Direct Prompting [21], and least-to-most [22]) and prominent retrieval-augmented approaches such as RAG [23], IRCoT [24], SKR [13], REPLUG [25], Self-RAG [12], and RA-ISF [6]. All evaluations use Exact Match (EM) as the primary metric, consistent with standard practices in retrieval-augmented question answering [6].

A. Llama2-13B Evaluation

Table 1 reports results evaluated on Llama2-13B across four datasets. We also present an average EM score to capture overall performance.

TABLE I. Exact Match Scores Evaluated on Llama2-13B on Four QA Benchmarks

Method	Avg	NQ	TriviaQA	HotpotQA	StrategyQA
Vanilla LM	30.53	17.4	38.5	14.0	52.2
Least-to-most	36.08	22.8	45.2	15.8	60.5
IRCoT	36.97	23.4	48.3	17.1	59.1
RAG	36.75	21.6	47.0	17.6	60.8
SKR_{knn}	39.17	20.8	55.4	18.9	61.6
REPLUG	41.77	23.8	58.6	21.8	62.9
Self-RAG_{13B}	47.58	28.4	69.3	25.4	67.2
RA-ISF	49.58	**31.3**	71.4	28.9	66.7
STAR (Ours)	**50.98**	31.0	**71.9**	**30.0**	**71.0**

We noticed that vanilla LM yields an average EM of 30.53, indicating that direct prompting without any retrieval or chain-of-thought struggles on knowledge-intensive queries. Approaches such as *Least-to-Most* (36.08 EM) and *RAG* (36.75 EM) provide moderate gains by introducing basic retrieval or prompt decomposition. *RA-ISF* achieves a strong baseline at 49.58 EM, reflecting its robust iterative method. In contrast, our two-branch architecture obtains 50.98 EM on average, surpassing both iterative and single-pass retrieval methods. The results confirm that avoiding unnecessary decomposition can reduce overhead while still maintaining or improving EM scores.

B. GPT-3.5 Evaluation

We next present evaluations on GPT-3.5, summarized in Table 2. Again, we compare our approach with multiple baselines. Direct prompting obtains an average of 45.95 EM, showing that GPT-3.5 alone does reasonably well but still misses specialized knowledge. RA-ISF yields a high average of 59.68 EM, confirming that iterative decomposition can boost performance significantly. Our method achieves 58.63 EM overall, slightly below RA-ISF on average, but outperforming or matching it on specific tasks like TriviaQA (at 76.2 vs. 76.1).

TABLE II. Exact Match Scores Evaluated on GPT-3.5 on Four QA Benchmarks

Method	Avg	NQ	TriviaQA	HotpotQA	StrategyQA
Direct	45.95	29.2	67.3	22.1	65.2
Least-to-most	50.00	32.5	68.8	30.2	68.5
IRCoT	50.32	32.9	66.8	33.7	67.9
RAG	48.20	31.7	64.2	32.2	64.7
SKR_{knn}	51.40	33.8	67.5	34.2	70.1
RA-ISF	**59.68**	40.2	76.1	**46.5**	**75.9**
STAR (ours)	58.63	**42.9**	**76.2**	39.5	**75.9**

V. Discussion

For TriviaQA, both Llama2-13B and GPT-3.5 benefit significantly from retrieval. Our method consistently matches or outperforms other strong baselines, highlighting the efficacy of focused chunking and relevance scoring. For HotpotQA, Multi-hop reasoning tasks often favor iterative strategies, yet we still achieve near-state-of-the-art performance by aggressively filtering irrelevant text and leveraging the model's reasoning capabilities (Figure 3). On strategy-based questions (StrategyQA), self-knowledge checking and targeted retrieval reduces potential hallucinations, leading to better factual correctness. Although methods like RA-ISF slightly exceed our performance on GPT-3.5's average EM, as shown in Figure 4, they may incur additional retrieval calls or iterative question breakdowns. In contrast, our solution remains consistently strong across tasks with minimal overhead, particularly benefiting real-time applications or token-limited scenarios. These experiments confirm that selectively retrieving external information (rather than always decomposing queries) can

deliver competitive or superior performance across diverse benchmarks.

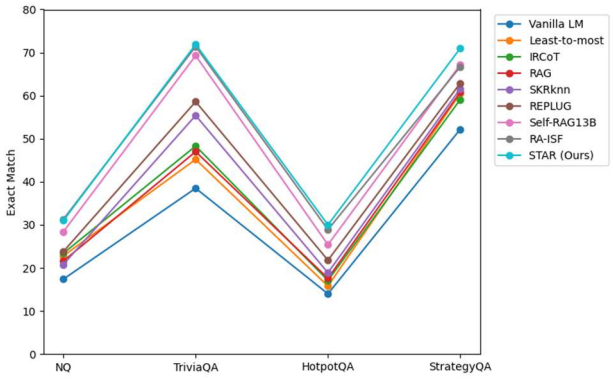

Fig. 3. Evaluation on Llama2-13B across Datasets

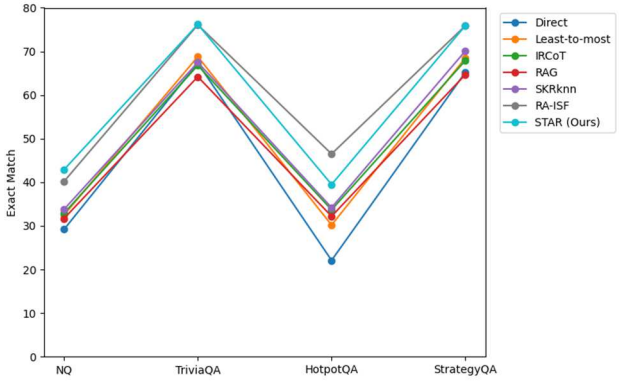

Fig. 4. Evaluation on GPT-3.5 across Datasets

VI. Conclusion and Outlook

We proposed a streamlined two-branch retrieval-augmented architecture that addresses knowledge gaps in large language models without resorting to iterative, multi-step decomposition. We combine a self-knowledge assessment with a focused retrieval pipeline to reduce computational overhead and irrelevant context. Experimental results on multiple benchmarks, using GPT-3.5 and Llama2-13B, confirm that this design maintains or exceeds the performance of more complex iterative frameworks, particularly on tasks where unnecessary decomposition can inflate runtime costs. In future research, we plan to extend this system's applicability beyond open-domain question answering to more specialized domains where efficient retrieval is critical, such as legal and medical texts. Additional efforts will explore further refining the passage segmentation strategy and integrating adaptive retrieval thresholds to support real-time scenarios. Privacy concerns about LLMs have surfaced, including memorization and jailbreaks [26], making it a priority to study the privacy implications of RAG-based approaches. Because STAR retrieves only narrowly relevant passages and skips unnecessary decomposition, it introduces less external text per query and may reduce memorization pressure and inadvertent leakage of sensitive content. Finally, we aim to investigate integrating advanced summarization or verification modules to further mitigate hallucination.

References

[1] S. Shahriar and R. Dara, "Priv-IQ: A Benchmark and Comparative Evaluation of Large Multimodal Models on Privacy Competencies," AI, vol. 6, no. 2, Art. no. 2, Feb. 2025, doi: 10.3390/ai6020029.

[2] Z. Zheng, C. Malon, M. R. Min, and X. Zhu, "Exploring the Role of Reasoning Structures for Constructing Proofs in Multi-Step Natural Language Reasoning with Large Language Models," in Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing, Y. Al-Onaizan, M. Bansal, and Y.-N. Chen, Eds., Miami, Florida, USA: Association for Computational Linguistics, Nov. 2024, pp. 15299–15312. doi: 10.18653/v1/2024.emnlp-main.854.

[3] S. Hao et al., "LLM Reasoners: New Evaluation, Library, and Analysis of Step-by-Step Reasoning with Large Language Models," presented at the First Conference on Language Modeling, Aug. 2024. Accessed: Feb. 25, 2025. [Online]. Available: https://openreview.net/forum?id=b0y6fbSUG0#discussion

[4] S. Shahriar et al., "Putting GPT-4o to the Sword: A Comprehensive Evaluation of Language, Vision, Speech, and Multimodal Proficiency," Appl. Sci., vol. 14, no. 17, Art. no. 17, Jan. 2024, doi: 10.3390/app14177782.

[5] P. Lewis et al., "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," in Advances in Neural Information Processing Systems, Curran Associates, Inc., 2020, pp. 9459–9474. Accessed: Feb. 25, 2025. [Online]. Available: https://proceedings.neurips.cc/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html

[6] Y. Liu et al., "RA-ISF: Learning to Answer and Understand from Retrieval Augmentation via Iterative Self-Feedback," Jun. 06, 2024, arXiv: arXiv:2403.06840. doi: 10.48550/arXiv.2403.06840.

[7] B. Dhingra, J. R. Cole, J. M. Eisenschlos, D. Gillick, J. Eisenstein, and W. W. Cohen, "Time-Aware Language Models as Temporal Knowledge Bases," Trans. Assoc. Comput. Linguist., vol. 10, pp. 257–273, Mar. 2022, doi: 10.1162/tacl_a_00459.

[8] W. Fan et al., "A Survey on RAG Meeting LLMs: Towards Retrieval-Augmented Large Language Models," in Proceedings of the 30th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, in KDD '24. New York, NY, USA: Association for Computing Machinery, Aug. 2024, pp. 6491–6501. doi: 10.1145/3637528.3671470.

[9] R. Ren et al., "Investigating the Factual Knowledge Boundary of Large Language Models with Retrieval Augmentation," Nov. 19, 2024, arXiv: arXiv:2307.11019. doi: 10.48550/arXiv.2307.11019.

[10] M. Skipanes, T. E. JÃ rgensen, K. Porter, G. Demartini, and S. Y. Yayilgan, "Enhancing Criminal Investigation Analysis with Summarization and Memory-based Retrieval-Augmented Generation: A Comprehensive Evaluation of Real Case Data," in Proceedings of the 31st International Conference on Computational Linguistics, O. Rambow, L. Wanner, M. Apidianaki, H. Al-Khalifa, B. D. Eugenio, and S. Schockaert, Eds., Abu Dhabi, UAE: Association for Computational Linguistics, Jan. 2025, pp. 4993–5010. Accessed: Feb. 20, 2025. [Online]. Available: https://aclanthology.org/2025.coling-main.334/

[11] F. Shi et al., "Large Language Models Can Be Easily Distracted by Irrelevant Context," in Proceedings of the 40th International Conference on Machine Learning, PMLR, Jul. 2023, pp. 31210–31227. Accessed: Feb. 20, 2025. [Online]. Available: https://proceedings.mlr.press/v202/shi23a.html

[12] A. Asai, Z. Wu, Y. Wang, A. Sil, and H. Hajishirzi, "Self-RAG: Learning to Retrieve, Generate, and Critique through Self-Reflection," presented at the The Twelfth International Conference on Learning Representations, Oct. 2023. Accessed: Feb. 20, 2025. [Online]. Available: https://openreview.net/forum?id=hSyW5go0v8

[13] Y. Wang, P. Li, M. Sun, and Y. Liu, "Self-Knowledge Guided Retrieval Augmentation for Large Language Models," in Findings of the Association for Computational Linguistics: EMNLP 2023, H. Bouamor, J. Pino, and K. Bali, Eds., Singapore: Association for Computational Linguistics, Dec. 2023, pp. 10303–10315. doi: 10.18653/v1/2023.findings-emnlp.691.

[14] J. Zhou, Z. Zheng, Y. Lyu, and T. Xu, "Enhancing Complex Question Answering via LLM Pseudo-Document and Adaptive Retrieval," in Web

Information Systems Engineering – WISE 2024, M. Barhamgi, H. Wang, and X. Wang, Eds., Singapore: Springer Nature, 2025, pp. 259–269. doi: 10.1007/978-981-96-0579-8_19.

[15] N. Matsumoto et al., "KRAGEN: a knowledge graph-enhanced RAG framework for biomedical problem solving using large language models," Bioinformatics, vol. 40, no. 6, p. btae353, Jun. 2024, doi: 10.1093/bioinformatics/btae353.

[16] G. Izacard et al., "Unsupervised Dense Information Retrieval with Contrastive Learning," Trans. Mach. Learn. Res., May 2022, Accessed: Feb. 25, 2025. [Online]. Available: https://openreview.net/forum?id=jKN1pXi7b0

[17] T. Kwiatkowski et al., "Natural Questions: A Benchmark for Question Answering Research," Trans. Assoc. Comput. Linguist., vol. 7, pp. 453–466, Aug. 2019, doi: 10.1162/tacl_a_00276.

[18] M. Joshi, E. Choi, D. S. Weld, and L. Zettlemoyer, "TriviaQA: A Large Scale Distantly Supervised Challenge Dataset for Reading Comprehension," May 13, 2017, arXiv: arXiv:1705.03551. doi: 10.48550/arXiv.1705.03551.

[19] Z. Yang et al., "HotpotQA: A Dataset for Diverse, Explainable Multi-hop Question Answering," Sep. 25, 2018, arXiv: arXiv:1809.09600. doi: 10.48550/arXiv.1809.09600.

[20] M. Geva, D. Khashabi, E. Segal, T. Khot, D. Roth, and J. Berant, "Did Aristotle Use a Laptop? A Question Answering Benchmark with Implicit Reasoning Strategies," Trans. Assoc. Comput. Linguist., vol. 9, pp. 346–361, Apr. 2021, doi: 10.1162/tacl_a_00370.

[21] T. Brown et al., "Language Models are Few-Shot Learners," in Advances in Neural Information Processing Systems, Curran Associates, Inc., 2020, pp. 1877–1901. Accessed: Feb. 25, 2025. [Online]. Available: https://proceedings.neurips.cc/paper/2020/hash/1457c0d6bfcb4967418bfb8ac142f64a-Abstract.html

[22] D. Zhou et al., "Least-to-Most Prompting Enables Complex Reasoning in Large Language Models," Apr. 16, 2023, arXiv: arXiv:2205.10625. doi: 10.48550/arXiv.2205.10625.

[23] K. Guu, K. Lee, Z. Tung, P. Pasupat, and M. Chang, "Retrieval Augmented Language Model Pre-Training," in Proceedings of the 37th International Conference on Machine Learning, PMLR, Nov. 2020, pp. 3929–3938. Accessed: Feb. 25, 2025. [Online]. Available: https://proceedings.mlr.press/v119/guu20a.html

[24] H. Trivedi, N. Balasubramanian, T. Khot, and A. Sabharwal, "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), A. Rogers, J. Boyd-Graber, and N. Okazaki, Eds., Toronto, Canada: Association for Computational Linguistics, Jul. 2023, pp. 10014–10037. doi: 10.18653/v1/2023.acl-long.557.

[25] W. Shi et al., "REPLUG: Retrieval-Augmented Black-Box Language Models," in Proceedings of the 2024 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (Volume 1: Long Papers), K. Duh, H. Gomez, and S. Bethard, Eds., Mexico City, Mexico: Association for Computational Linguistics, Jun. 2024, pp. 8371–8384. doi: 10.18653/v1/2024.naacl-long.463.

[26] S. Shahriar, R. Dara, and R. Akalu, "A comprehensive review of current trends, challenges, and opportunities in text data privacy," Comput. Secur., vol. 151, p. 104358, Apr. 2025, doi: 10.1016/j.cose.2025.104358.