Exploring the Integration Strategies of Retriever and Large Language Models

Preprint · August 2023

DOI: 10.48550/arXiv.2308.12574

Ye Liu
University of Illinois Chicago

Rui Meng
University of Pittsburgh

Shafiq Joty
University of British Columbia

Yingbo Zhou
University at Buffalo, State University of New York

Exploring the Integration Strategies of Retriever and Large Language Models

Ye Liu, Semih Yavuz, Rui Meng, Meghana Moorthy, Shafiq Joty, Caiming Xiong, Yingbo Zhou
Salesforce Research
{yeliu, syavuz, ruimeng, meghana.bhat, sjoty, cxiong, yingbo.zhou}@salesforce.com

Abstract

The integration of retrieved passages and large language models (LLMs), such as ChatGPTs, has significantly contributed to improving open-domain question answering. However, there is still a lack of exploration regarding the optimal approach for incorporating retrieved passages into the answer generation process. This paper aims to fill this gap by investigating different methods of combining retrieved passages with LLMs to enhance answer generation. We begin by examining the limitations of a commonly-used concatenation approach. Surprisingly, this approach often results in generating "unknown" outputs, even when the correct document is among the top-k retrieved passages. To address this issue, we explore four alternative strategies for integrating the retrieved passages with the LLMs. These strategies include two single-round methods that utilize chain-of-thought reasoning and two multi-round strategies that incorporate feedback loops. Through comprehensive analyses and experiments, we provide insightful observations on how to effectively leverage retrieved passages to enhance the answer generation capability of LLMs.

1 Introduction

Large Language Models (LLMs), such as GPTs (Brown et al., 2020; Bubeck et al., 2023), have found extensive applications, but often struggle with limited knowledge representation, resulting in inaccuracies and insufficient specificity in open-domain question answering. To overcome these limitations, the integration of retrieval-based techniques (Izacard et al., 2022; Borgeaud et al., 2022) has emerged as a promising solution. By incorporating relevant passages during the answer generation, LLMs can leverage external information to provide more accurate and detailed responses. Nevertheless, effective strategies for incorporating retrieved passages into the LLMs remains a challenging and relatively understudied area.

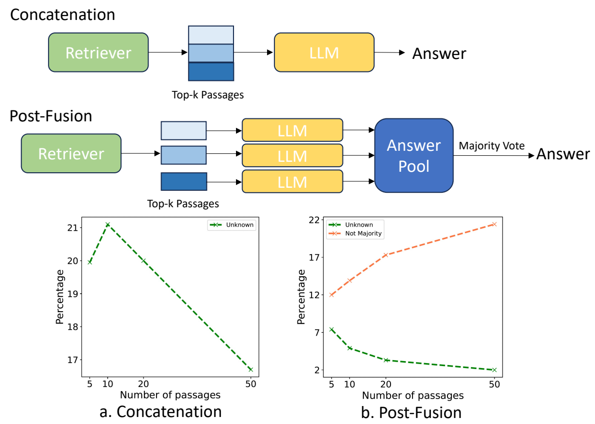

Figure 1: **Top**: Illustration of Concatenation v.s. Post-Fusion strategies. **Bottom-a**: percentage of unknown responses using the Concatenation strategy. **Bottom-b**: by varying the number of retrieved passages, (green) percentage of unknown responses, and (red) error rate by majority voting (when the correct answer is in the answer pool, the majority selects a wrong answer).

Our analysis (Fig. 1), conducted under the oracle setting where one of the top-k retrieved passages contains the answer, reveals that a simple concatenation of the passages into LLMs often leads to "unknown" responses — instances where the provided context fails to answer the question — accounting for about 20% of all responses. An alternative method, where the passages are individually provided as input to LLMs and the majority vote determines the final answer, reduces the rate of "unknown" generation to 2-7% depending on the number of passages. However, this method introduces a new challenge: the correct answer does not align with the majority vote in the answer pool. Particularly, when more passages are incorporated from 5 to 50, the error rate of the majority vote increases from 12% to 22%. Thus, both of the methods have their own weaknesses and more suitable approaches for the integration of retrieved passages and LLMs remain to be investigated.

Transformer-based LLMs have shown the capability to utilize attention mechanisms (Vaswani et al., 2017) for discovering token-level relevance. However, they may not always attend to the relevant parts within the context, leading to a potential oversight of important information present in the retrieved passages (Clark et al., 2020; Zhao et al., 2020). This challenge becomes more pronounced when dealing with extensive corpora like Wikipedia, which contains over 21 million passages, making it a formidable task to identify the most relevant passages for a question. Furthermore, retrieved passages that are closely related to the question's topic can act as distractors, potentially misleading the model (Asai et al., 2019). If the model mistakenly directs its attention towards these distractor passages, it can introduce noise that negatively impacts the answer prediction process.

In this paper, we explore the integration of retrieved passages with LLMs like ChatGPTs to enhance their ability to generate correct answers. In particular, we examine situations where the retrieved passages contain the correct answer, yet the model fails to generate the correct response, indicating areas for improvement. Initially, we focus on two chain-of-thought (CoT) (Wei et al., 2022) strategies that incorporate in-context learning. We introduce a pruning strategy and a summarization strategy for the retrieved passages to guide the answer generation process of the LLMs. Subsequently, we investigate two feedback methods: presenting the retrieved passages to the LLMs, collecting its responses, and then modifying the LLM interaction based on this feedback.

Through a series of experiments on three single-hop open-domain question answering datasets, we demonstrate the effectiveness of the proposed approaches. Our findings provide a foundation for the development of more advanced retrieval-integration methods aimed at further enhancing the capabilities of these models.

2 Problem Setup

This study focuses on the question answering task under the open-domain setting. It remains a open problem to retrieve the most relevant context for question answering. Therefore, a common practice is to include multiple top ranked passages, which serves as the supplementary context for the LLMs. The number of supplementary passages, denoted as k, can vary based on the desired input length M of the LLM. Typically, k can be set to 5, 10, or 20, ensuring that the total length of k passages, each having a maximum length of L, remains within the maximum input length M of the LLM (i.e., $k * L < M$). By incorporating these supplementary passages, the LLM is provided with a more comprehensive and informative context, which has the potential to enhance its accuracy.

3 Methods

We adopt a two-stage pipeline for open-domain QA. It consists of two black-box components, a retriever and a LLM such as ChatGPT. We aim to methodically investigate the optimal methods for transferring the top-k retrieval results to the LLMs for generating factoid answers. Our investigation begins with a focus on various **single-round** strategies, wherein the retrieved passages are directly fed into the LLMs. Subsequently, we delve into several **multi-round** approaches, involving the initial supply of retrieved passages to the LLMs, gathering feedback, and then modifying the interaction process with the LLMs based on that feedback.

3.1 Single-Round Approaches

We introduce two distinct prompts, with one-shot example, to guide the LLMs in fusing answers from potentially relevant passages. Examples of these two prompt types are provided in Fig. 6 and 7 in the appendix, respectively.

Pruning Prompt. This prompt requires the LLM to effectively identify answerable passages through a process of elimination. As a result, The demonstration involves differentiating irrelevant passages from the ones that can provide an answer, and subsequently generating the final response.

Summary Prompt. Summarization strategy condenses core information from the top-k passages to guide LLMs in formulating responses (Demner-Fushman and Lin, 2006). A demonstration example illustrates this process is provided, showcasing how the model should extract useful information before generating answers.

3.2 Multi-Round Approaches

In our exploration of multi-round strategies, we first provide the retrieved passages to the LLM. Based on the initial feedback received, we adjust our interaction process with the LLM accordingly.

Post-Fusion as the Fallback (Concat+PF). Initially, we employ the concatenation method as illustrated in Fig. 2 to obtain an answer predicted by the LLM. If the LLM determines that the in-

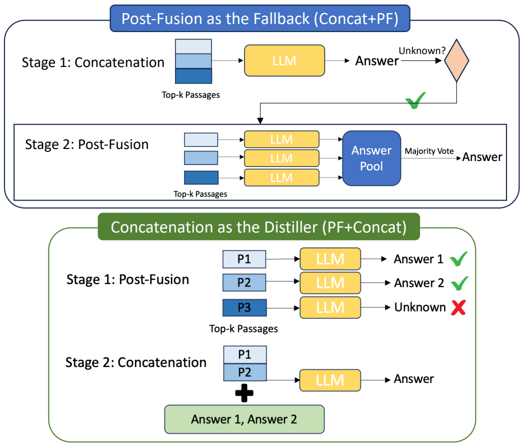

Figure 2: Diagram of Post-Fusion as the Fallback on top and Concatenation as the Distiller at bottom.

put passages are unable to provide an answer to the question (i.e., "unknown" responses), we then proceed to the second round where we utilize the Post-Fusion approach to produce an answer pool. Finally, we employ a majority vote to select the final answer.

Concatenation as the Distiller (PF+Concat). To begin with, we leverage the Post-Fusion strategy to curate a pool of potential answers shown in Fig. 2. Instead of performing a majority vote, a passage selection process (Lewis et al., 2020) is adopted to discard passages that yield an "unknown" output by the LLM. In the second round, the LLM is prompted with the concatenation of the unfiltered passages, along with the question and answer candidates generated from the first round. The purpose is to guide the LLM in effectively extracting (distilling) the correct answer from the pool of candidates.

4 Experiments

Evaluation Benchmarks. We conduct evaluations on multiple datasets of open-domain question answering to assess the performance of the proposed integration approaches.

The datasets used include Natural Questions (NQ) (Yang et al., 2018), TriviaQA (Trivedi et al., 2022b), and SQuAD-Open (Ho et al., 2020) are all datasets designed for training and evaluating single-hop question answering models. NQ is sourced from Google Search queries and their corresponding Wikipedia answers. TriviaQA offers a broader domain with trivia questions and their answers derived from web and Wikipedia sources. Conversely, SQuAD-Open is a variant of the original SQuAD dataset that requires the model to extract answers from open-domain Wikipedia content, without any pre-specified passage.

Predicted answers are evaluated with the standard exact match (EM) and F1 metric (Rajpurkar et al., 2016; Liu et al., 2022). A generated response is considered correct if, after normalization, it matches any candidate in a list of acceptable answers. The normalization process entails converting the text to lowercase and omitting articles, punctuation, and redundant whitespaces. We also evaluate the percentage of unknown responses (%Unk) and the error rate by majority vote (%NM).

Dataset Filter To mitigate the influence of specific training datasets on the LLM (Aiyappa et al., 2023), we initially prompt the LLM to answer questions without any provided context. This process enables us to filter out questions that the LLM can accurately answer independently, thereby eliminating the need for additional external contextual information. The remaining questions, which the LLM couldn't answer independently, are the focus of our study. This filtering ensures our evaluation stringently reflects the LLM's ability to utilize external context from retrieved passages.

We use the development set of NQ, TriviaQA, and SQuAD, initially containing 5,892, 6,760, 5,928 questions, respectively. After removing questions that can be answered without context, we are left with 3,459 questions in NQ, 1,259 in TriviaQA, and 3,448 in SQuAD.

Retriever and LLM model. We use the Wikipedia dump from Dec. 20, 2018 for NQ and TriviaQA and the dump from Dec. 21, 2016 for SQuAD. We apply preprocessing steps following Chen et al. (2017); Karpukhin et al. (2020); Liu et al. (2021), which involve generating non-overlapping passages of 100 words each. Similar to (Izacard and Grave, 2020), passages are retrieved with DPR (Karpukhin et al., 2020) for NQ and TriviaQA and with BM25 (Robertson et al., 1995) for SQuAD. We consider two different settings for this study. The first utilizes the top-k retrieved passages directly (gold passage is not necessarily included). In contrast, the second setting concerns the situation that the gold-standard passage is included in the context. If the gold passage is not within the top-k passages, we randomly insert it into the top-k list.

We choose the `gpt-3.5-turbo-16k` by OpenAI as our LLM, and we perform greedy decoding by setting the temperature parameter to 0.

	NQ				TriviaQA				SQuAD			
	EM	F1	%Unk	%NM	EM	F1	%Unk	%NM	EM	F1	%Unk	%NM
Supervised	40.9	-	-	-	55.2	-	-	-	35.8	-	-	-
Without gold passage												
Concatenation	34.5	43.8	23.1%	-	49.3	55.5	19.9%	-	28.1	34.8	28.5%	-
Post-Fusion	38.3	48.3	10.1%	9.0%	49.7	55.7	10.7%	7.4%	32.1	40.3	13.9%	12.3%
Pruning Prompt	36.2	46.3	9.1%	-	49.3	56.5	9.5%	-	36.1	40.6	12.7%	-
Summary Prompt	36.3	48.4	**8.6%**	-	48.3	56.5	**7.7%**	-	34.1	40.0	13.7%	-
Concat + PF	**39.9**	49.7	9.3%	1.1%	**52.7**	**59.5**	9.1%	0.7%	**40.1**	**43.8**	**5.7%**	2.2%
PF + Concat	38.9	**50.1**	10.1%	**0%**	50.5	57.7	10.7%	**0%**	38.5	41.2	13.9%	**0%**
With gold passage												
Concatenation	38.1	45.4	19.9%	-	51.6	57.9	18.1%	-	53.1	64.9	13.6%	-
Post-Fusion	40.1	50.4	7.4%	12.0%	51.4	57.3	9.1%	10.2%	57.1	71.2	2.1%	4.3%
Pruning Prompt	39.0	50.5	6.9%	-	52.7	59.5	8.1%	-	47.7	62.6	6.7%	-
Summary Prompt	40.5	53.3	**5.1%**	-	51.6	60.1	**6.4%**	-	50.4	67.04	4.7%	-
Concat + PF	42.9	53.9	6.5%	1.8%	**55.9**	**62.8**	7.5%	1.1%	60.6	74.0	**1.7%**	0.2%
PF + Concat	**43.2**	**54.5**	7.4%	**0%**	54.0	61.7	9.2%	**0%**	**63.9**	**76.9**	2.1%	**0%**

Table 1: Exact match (EM) and F1 scores on filtered DEV split of the NQ, TriviaQA and SQuAD using Top-5 passages. %Unk denotes the percentage of Unknown responses. %NM denotes the error rate by majority vote. **Concat** refers to the Concatenation strategy and **PF** refers to Post-Fusion strategy.

4.1 Results

As shown in Table 1, Post-Fusion strategies generally outperform concatenation approaches. Single-round methods, such as Pruning Prompt and Summary Prompt, exhibit better performance compared to the concatenation approach. The use of the CoT, which elicits a potential reasoning process, can guide the model in attending to relevant passages. However, this approach does not greatly enhance multi-hop question answering as compared to prior studies (Wei et al., 2022; Trivedi et al., 2022a).

In comparison to single-round methods, multi-round strategies tend to perform better overall, suggesting the efficacy of incorporating the model feedback. Among the multi-round approaches, Concat + PF demonstrates superior performance compared to PF + Concat. Comparing PF + Concat with Post-Fusion, it is evident that PF + Concat, leveraging LLM to select the best answer from a candidate pool, outperforms the majority vote approach. Our proposed strategies achieve comparable performance to supervised results, even without including a gold passage in the top-k.

4.2 Usage Analysis

Finding a balance between improving the quality of generated answers and resource utilization is crucial. Figure 3 illustrates the token usage of different approaches. The Concatenate method consumes the fewest resources, followed by Concat + PF as the second resource-efficient approach. Considering the performance advantage of Concat + PF over

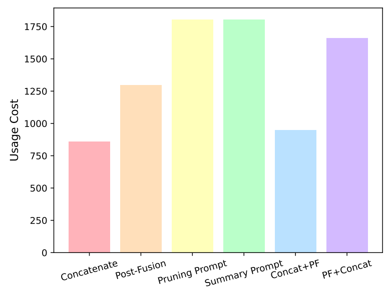

Figure 3: The token usage of different approaches using top-5 passages.

Concatenate (Table 1), we recommend employing Concat + PF as a more effective approach for integrating retrieved passages with LLMs.

5 Conclusion

In this study, we identified two key challenges associated with integrating LLMs and retrieved passages: the occurrence of "unknown" responses when feeding LLMs with concatenated passages and the erroneous majority when using the Post-Fusion approach. To overcome these challenges, we proposed four improved approaches, including two CoT-related strategies and two multi-round methods incorporating LLM's feedback. Through our experimental results and token usage analysis, we observed that it is advantageous to first employ a concatenation strategy to generate an answer. In the case of an "unknown" response, we recommend transitioning to the Post-Fusion approach to obtain the final answer through a majority vote.

Limitations

While our method is a general approach, we have so far only tested it on ChatGPT. In future, we plan to extend our evaluation to other LLMs like GPT-4 (OpenAI, 2023), Falcon (Penedo et al., 2023), LLaMA (Touvron et al., 2023) et al.

To have better comparison with supervised SOTA approach (Izacard and Grave, 2020), we only evaluate on three open-domain QA datasets. To further test our findings and the efficacy of proposed methods, we plan to evaluate them on additional open-domain single-hop question answering benchmarks, such as the MS MARCO, WebQuestions datasets (Nguyen et al., 2016; Berant et al., 2013).

References

Rachith Aiyappa, Jisun An, Haewoon Kwak, and Yong-Yeol Ahn. 2023. Can we trust the evaluation on chatgpt? *arXiv preprint arXiv:2303.12767*.

Akari Asai, Kazuma Hashimoto, Hannaneh Hajishirzi, Richard Socher, and Caiming Xiong. 2019. Learning to retrieve reasoning paths over wikipedia graph for question answering. *arXiv preprint arXiv:1911.10470*.

Jonathan Berant, Andrew Chou, Roy Frostig, and Percy Liang. 2013. Semantic parsing on freebase from question-answer pairs. In *Proceedings of the 2013 conference on empirical methods in natural language processing*, pages 1533–1544.

Sebastian Borgeaud, Arthur Mensch, Jordan Hoffmann, Trevor Cai, Eliza Rutherford, Katie Millican, George Bm Van Den Driessche, Jean-Baptiste Lespiau, Bogdan Damoc, Aidan Clark, et al. 2022. Improving language models by retrieving from trillions of tokens. In *International conference on machine learning*, pages 2206–2240. PMLR.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *Advances in neural information processing systems*, 33:1877–1901.

Sébastien Bubeck, Varun Chandrasekaran, Ronen Eldan, Johannes Gehrke, Eric Horvitz, Ece Kamar, Peter Lee, Yin Tat Lee, Yuanzhi Li, Scott Lundberg, et al. 2023. Sparks of artificial general intelligence: Early experiments with gpt-4. *arXiv preprint arXiv:2303.12712*.

Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. 2017. Reading wikipedia to answer open-domain questions. *arXiv preprint arXiv:1704.00051*.

Wenhu Chen, Pat Verga, Michiel de Jong, John Wieting, and William Cohen. 2022. Augmenting pre-trained language models with qa-memory for open-domain question answering. *arXiv preprint arXiv:2204.04581*.

Kevin Clark, Minh-Thang Luong, Quoc V Le, and Christopher D Manning. 2020. Electra: Pre-training text encoders as discriminators rather than generators. *arXiv preprint arXiv:2003.10555*.

Dina Demner-Fushman and Jimmy Lin. 2006. Answer extraction, semantic clustering, and extractive summarization for clinical question answering. In *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the Association for Computational Linguistics*, pages 841–848.

Luyu Gao, Xueguang Ma, Jimmy Lin, and Jamie Callan. 2022a. Precise zero-shot dense retrieval without relevance labels. *arXiv preprint arXiv:2212.10496*.

Luyu Gao, Aman Madaan, Shuyan Zhou, Uri Alon, Pengfei Liu, Yiming Yang, Jamie Callan, and Graham Neubig. 2022b. Pal: Program-aided language models. *arXiv preprint arXiv:2211.10435*.

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. 2020. Constructing a multi-hop qa dataset for comprehensive evaluation of reasoning steps. *arXiv preprint arXiv:2011.01060*.

Gautier Izacard and Edouard Grave. 2020. Leveraging passage retrieval with generative models for open domain question answering. *arXiv preprint arXiv:2007.01282*.

Gautier Izacard, Patrick Lewis, Maria Lomeli, Lucas Hosseini, Fabio Petroni, Timo Schick, Jane Dwivedi-Yu, Armand Joulin, Sebastian Riedel, and Edouard Grave. 2022. Few-shot learning with retrieval augmented language models. *arXiv preprint arXiv:2208.03299*.

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

Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. *arXiv preprint arXiv:2004.04906*.

Omar Khattab, Keshav Santhanam, Xiang Lisa Li, David Hall, Percy Liang, Christopher Potts, and Matei Zaharia. 2022. Demonstrate-search-predict: Composing retrieval and language models for knowledge-intensive nlp. *arXiv preprint arXiv:2212.14024*.

Angeliki Lazaridou, Elena Gribovskaya, Wojciech Stokowiec, and Nikolai Grigorev. 2022. Internet-augmented language models through few-shot prompting for open-domain question answering. *arXiv preprint arXiv:2203.05115*.

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. *Advances in Neural Information Processing Systems*, 33:9459–9474.

Ye Liu, Kazuma Hashimoto, Yingbo Zhou, Semih Yavuz, Caiming Xiong, and Philip S Yu. 2021. Dense hierarchical retrieval for open-domain question answering. *arXiv preprint arXiv:2110.15439*.

Ye Liu, Semih Yavuz, Rui Meng, Dragomir Radev, Caiming Xiong, and Yingbo Zhou. 2022. Uniparser: Unified semantic parser for question answering on knowledge base and database. *arXiv preprint arXiv:2211.05165*.

Yuning Mao, Pengcheng He, Xiaodong Liu, Yelong Shen, Jianfeng Gao, Jiawei Han, and Weizhu Chen. 2020. Generation-augmented retrieval for open-domain question answering. *arXiv preprint arXiv:2009.08553*.

Tri Nguyen, Mir Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, and Li Deng. 2016. Ms marco: A human generated machine reading comprehension dataset. *choice*, 2640:660.

OpenAI. 2023. Gpt-4 technical report.

Bhargavi Paranjape, Scott Lundberg, Sameer Singh, Hannaneh Hajishirzi, Luke Zettlemoyer, and Marco Tulio Ribeiro. 2023. Art: Automatic multi-step reasoning and tool-use for large language models. *arXiv preprint arXiv:2303.09014*.

Hyun Jin Park and Changwan Ryu. 2023. Query augmentation using search engine results to improve answers generated by large language models.

Guilherme Penedo, Quentin Malartic, Daniel Hesslow, Ruxandra Cojocaru, Alessandro Cappelli, Hamza Alobeidli, Baptiste Pannier, Ebtesam Almazrouei, and Julien Launay. 2023. The refinedweb dataset for falcon llm: outperforming curated corpora with web data, and web data only. *arXiv preprint arXiv:2306.01116*.

Ofir Press, Muru Zhang, Sewon Min, Ludwig Schmidt, Noah A Smith, and Mike Lewis. 2022. Measuring and narrowing the compositionality gap in language models. *arXiv preprint arXiv:2210.03350*.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*.

Stephen E Robertson, Steve Walker, Susan Jones, Micheline M Hancock-Beaulieu, Mike Gatford, et al. 1995. Okapi at trec-3. *Nist Special Publication Sp*, 109:109.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. 2023. Toolformer: Language models can teach themselves to use tools. *arXiv preprint arXiv:2302.04761*.

Thomas Scialom, Tuhin Chakrabarty, and Smaranda Muresan. 2022. Continual-t0: Progressively instructing 50+ tasks to language models without forgetting. *arXiv preprint arXiv:2205.12393*.

Zhihong Shao, Yeyun Gong, Yelong Shen, Minlie Huang, Nan Duan, and Weizhu Chen. 2023. Synthetic prompting: Generating chain-of-thought demonstrations for large language models. *arXiv preprint arXiv:2302.00618*.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, et al. 2023. Llama: Open and efficient foundation language models. *arXiv preprint arXiv:2302.13971*.

Harsh Trivedi, Niranjan Balasubramanian, Tushar Khot, and Ashish Sabharwal. 2022a. Interleaving retrieval with chain-of-thought reasoning for knowledge-intensive multi-step questions. *arXiv preprint arXiv:2212.10509*.

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

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Ed Chi, Quoc Le, and Denny Zhou. 2022. Chain of thought prompting elicits reasoning in large language models. *arXiv preprint arXiv:2201.11903*.

Zhilin Yang, Peng Qi, Saizheng Zhang, Yoshua Bengio, William W Cohen, Ruslan Salakhutdinov, and Christopher D Manning. 2018. Hotpotqa: A dataset for diverse, explainable multi-hop question answering. *arXiv preprint arXiv:1809.09600*.

Shunyu Yao, Jeffrey Zhao, Dian Yu, Nan Du, Izhak Shafran, Karthik Narasimhan, and Yuan Cao. 2022. React: Synergizing reasoning and acting in language models. *arXiv preprint arXiv:2210.03629*.

Jingfeng Zhang, Haiwen Hong, Yin Zhang, Yao Wan, Ye Liu, and Yulei Sui. 2021. Disentangled code representation learning for multiple programming languages. *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*.

Chen Zhao, Chenyan Xiong, Corby Rosset, Xia Song, Paul Bennett, and Saurabh Tiwary. 2020. Transformer-xh: Multi-evidence reasoning with extra hop attention.

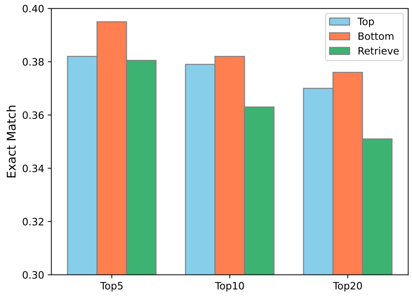

Figure 4: The impact on the position of gold passage on Combination method.

A Does the order of the gold passage influence LLM?

In this section, we aim to assess how the placement of the gold passage within the Top-k passages influences the ability of the LLM to generate accurate answers. We examine three different placements: (1) consistently positioning the gold passage at the start of the Top-k passage list; (2) consistently placing the gold passage at the end of the Top-k passage list; (3) maintaining the original sequence produced by the retrieval model.

As the results depicted in Fig. 4, it is evident that the placement of the gold passage significantly affects the quality of the generated answers. Consistently placing the gold passage in the same position tends to improve performance compared to using the retrieval order. Among the constant placement options, positioning the gold passage at the bottom tends to yield better results than placing it at the top. This outcome might be tied to our prompt design, where we present the Top-k passages first, followed by the question. Consequently, keeping the gold passage closer to the question seems to enhance performance to the greatest extent.

B Related Work

The recent proliferation of LLM-powered applications, such as ChatGPT/GPT4 (OpenAI, 2023), Bing Chat, and CoPilot, has highlighted both the impressive performance and certain limitations of LLMs. These limitations include a high compute and data demand, making it a challenge to continually update LLMs both efficiently and effectively (Scialom et al., 2022). LLMs also tend to generate plausible yet non-factual texts, a phenomenon known as 'hallucination' (OpenAI, 2023). In response to these issues, the field is witnessing a trend towards augmenting LLMs with specialized tools (Schick et al., 2023; Paranjape et al., 2023), such as code interpreters (Zhang et al., 2021; Gao et al., 2022b; Shao et al., 2023) or search engines (Park and Ryu, 2023). The goal is to delegate specific tasks to more proficient systems or to enrich the LLMs' input context with more pertinent information.

Augmentation of language models with pertinent data retrieved from diverse knowledge bases has demonstrated its effectiveness in enhancing open-domain question answering performance (Lazaridou et al., 2022; Izacard et al., 2022; Chen et al., 2022). The process typically involves using the input query to (1) command a retriever to fetch a document set (essentially, token sequences) from a corpus, after which (2) the language model integrates these retrieved documents as supplemental information, guiding the final prediction.

The interleaving between the retriever and LLM could be considered a reciprocal process. Various studies have been conducted on generation-augmented retrieval (GAR), which involves revising or supplementing queries with generated background information to enhance the retrieval of relevant content. Well-known examples of this approach include GAR (Mao et al., 2020) and HyDE (Gao et al., 2022a). With regard to complex multi-step reasoning questions, work involving LLMs often necessitates the retrieval of segmented knowledge (Trivedi et al., 2022a; Khattab et al., 2022). This chain-of-thought reasoning process (Wei et al., 2022; Jiang et al., 2023) is followed by conducting partial reasoning to generate the next question, then retrieving further information based on the outcome of that partially formed next question, and repeating this cycle as needed (Yao et al., 2022; Press et al., 2022).

Our work primarily focuses on a specific scope: once the output from the retriever is determined, we aim to identify the most effective method of inputting this data into LLMs for answer generation.

C Prompt used in Different Approaches

The prompts used in the Concatenation and Post-Fusion approaches are illustrated in Fig. 5. The Pruning Prompt's specific prompt is presented in Fig. 6, while the Summary Prompt's prompt is depicted in Fig. 7.

Given the relevant background contexts, answer the current question using one of the context in short factoid phrase manner.

Question: Who produced the album that included a re-recording of \"Lithium\"?
Answer: Butch Vig
Question: What city was the victim of Joseph Druces working in?
Answer: Boston, Massachusetts
Question: In what year was the star of To Hell and Back born?
Answer: 1925

Try your best to guess an extractive answer. If don't know the answer, just say **unknown**.
Context:
{retrieved_context}
Question:
{question}
Answer:

Figure 5: The Prompt used in Concatenation and Post-Fusion.

Answer questions with short factoid answers.

Question: Who produced the album that included a re-recording of \"Lithium\"?
Answer: Butch Vig
Question: What city was the victim of Joseph Druces working in?
Answer: Boston, Massachusetts
Question: In what year was the star of To Hell and Back born?
Answer: 1925

Follow the following format.
Context:
sources that may contain relevant content
Question:
the question to be answered
Rationale: Let's think step by step. a step-by-step deduction that identifies the correct response, which will be provided below
Answer: a short factoid answer, often between 1 and 5 words. Make sure generate \"Answer\": in the end!
If don't know the answer, just say **unknown** as answer.

Context:
[1] Peter Outerbridge | Peter Outerbridge Peter Outerbridge (born June 30, 1966) is a Canadian actor…..
[2] Except the Dying | 2008. On March 3, 2015, Acorn Media announced a re-release for all three movies, set for May 26, 2015…..
[3] «Saw VI | Saw VI Saw VI is a 2009 American horror film directed by Kevin Greutert from a screenplay written by Patrick Melton and Marcus Dunstan. It is the sixth installment in the \"Saw\" franchise and stars Tobin Bell……
Question: Which 2009 movie does Peter Outerbridge feature as William Easton?
Rationale: Let's think step by step.
The question is asking for the 2009 movie that Peter Outerbridge was in as William Easton. We can use process of **pruning** to figure this out. Source 1 doesn't contain the information. In source 2, it talks about a made-for-TV movie in 2004. In source 3, it talks about the sixth installment in the \"Saw\" franchise. This must be the movie we are looking for.
Answer:
Saw VI

Context:
{retrieved_topk_context}
Question:
{question}
Rationale: Let's think step by step.

Figure 6: The Pruning Prompt.

Answer questions with short factoid answers.

Question: Who produced the album that included a re-recording of \"Lithium\"?
Answer: Butch Vig
Question: What city was the victim of Joseph Druces working in?
Answer: Boston, Massachusetts
Question: In what year was the star of To Hell and Back born?
Answer: 1925

Follow the following format.
Context:
sources that may contain relevant content
Question:
the question to be answered
Rationale: Let's think step by step. a step-by-step deduction that identifies the correct response, which will be provided below
Answer: a short factoid answer, often between 1 and 5 words. Make sure generate \"Answer\": in the end!
If don't know the answer, just say **unknown** as answer.

Context:
[1] Peter Outerbridge | Peter Outerbridge Peter Outerbridge (born June 30, 1966) is a Canadian actor…..
[2] Except the Dying | 2008. On March 3, 2015, Acorn Media announced a re-release for all three movies, set for May 26, 2015…..
[3] «Saw VI | Saw VI Saw VI is a 2009 American horror film directed by Kevin Greutert from a screenplay written by Patrick Melton and Marcus Dunstan. It is the sixth installment in the \"Saw\" franchise and stars Tobin Bell……
Question: Which 2009 movie does Peter Outerbridge feature as William Easton?
Rationale: Let's think step by step.
The question requires information on the 2009 movie that Peter Outerbridge was in as William Easton. Going through the provided sources, we can narrow down our focus to Source 3 and Source 4 that mention \"Saw VI\", a movie released in 2009, in which Peter Outerbridge starred. By **summarizing** these details, the movie from 2009 featuring Peter Outerbridge is \"Saw VI\".
Answer:
Saw VI

Context:
{retrieved_topk_context}
Question:
{question}
Rationale: Let's think step by step.

Figure 7: The Summary Prompt.