Letriever: Using Large Language Models as Contextual Retriever for Long-Context Question Answering

Anonymous ACL submission

Abstract

Large Language Models (LLMs) have demonstrated outstanding performance on Question Answering (QA) tasks. However, they face significant challenges in long-context QA due to difficulties in effectively utilizing lengthy inputs, resulting in irrelevant responses. While Retrieval-Augmented Generation (RAG) frameworks have been employed to address this issue, they remain limited by retrieval methods that prioritize superficial lexical overlaps, leading to suboptimal context selection. In this study, we propose *Letriever*, which replaces the traditional embedding-based retriever with an LLM-based retriever. By leveraging the advanced comprehension capabilities of LLMs, *Letriever* enhances retrieval precision and answer accuracy across diverse QA benchmarks. Our findings highlight the potential of LLMs to transform retrieval mechanisms in QA systems.

1 Introduction

The growing complexity of real-world QA tasks highlights the need for methods that can effectively process and reason over long, information-rich contexts. With the enhanced token capacity of Large Language Models (LLMs) (Achiam et al., 2023; Anthropic, 2024) have shown impressive performance across various QA tasks. However, studies (Shi et al., 2023) reveal that excessively long contexts can introduce noise, making it challenging for models to accurately identify and reference relevant information.

Retrieval-Augmented Generation (RAG) is commonly employed to address this issue. By leveraging semantic search methods such as cosine similarity, RAG filters out irrelevant noise by assuming that the most similar content is also the most relevant. The limitation of this approach is that it may overlook semantically relevant information expressed in different ways, leading to inaccuracies in evidence retrieval and downstream answer generation.

To address this issue, we propose Letriever, which replaces the traditional embedding-based retriever with an LLM-based model. Our method leverages the advanced natural language reasoning capabilities of LLMs in an end-to-end manner, where they function as both retrievers and generators. Instead of processing the entire long context, our approach extracts the most relevant information based on the query. Unlike traditional RAG, our method can understand complex natural language information and ensure robust performance by reducing dependency on hyperparameters such as top-k selection.

We summarize our contributions as follows:

- We propose an LLM-based retriever approach that replaces traditional cosine similarity-based semantic search in Retrieval-Augmented Generation (RAG).
- We conduct experiments across diverse QA datasets, demonstrating that the contextual understanding of LLMs enables a more nuanced representation of complex semantics.
- We show that leveraging the reasoning capabilities of LLMs allows for flexible adjustment of hyperparameters tailored to each dataset, yielding robust performance across diverse QA tasks.

This paper addresses the limitations of existing RAG methods and explores a new approach to leverage LLMs as a retriever in QA tasks. Our research aims to enhance the accuracy of key information retrieval while minimizing noise, making it particularly effective for long documents such as extended conversations or technical papers.

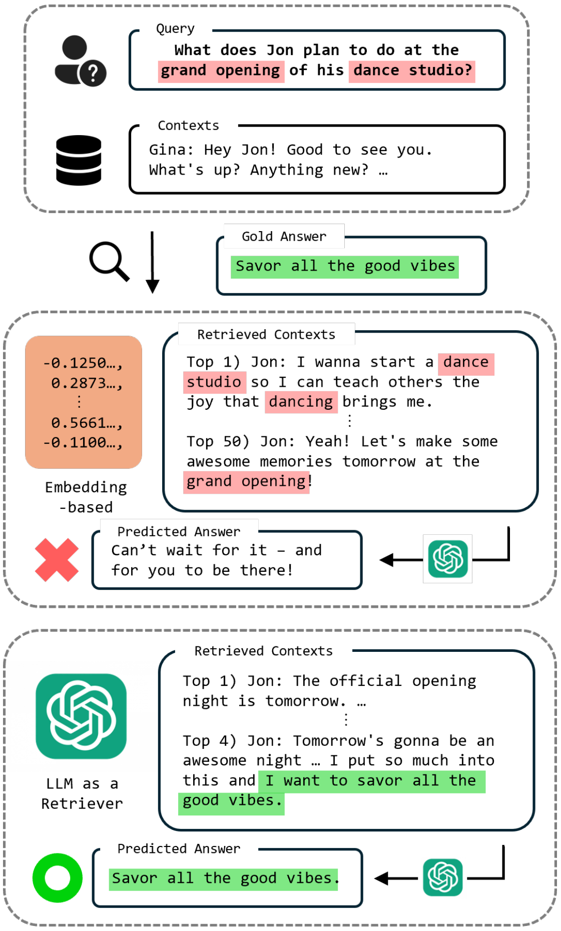

Query
What does Jon plan to do at the grand opening of his dance studio?
Contexts
Gina: Hey Jon! Good to see you. What's up? Anything new? …
Gold Answer
Savor all the good vibes
-0.1250…,
0.2873…,
⋮
0.5661…,
-0.1100…,
Embedding -based
Retrieved Contexts
Top 1) Jon: I wanna start a dance studio so I can teach others the joy that dancing brings me.
⋮
Top 50) Jon: Yeah! Let's make some awesome memories tomorrow at the grand opening!
Predicted Answer
Can't wait for it – and for you to be there!
LLM as a Retriever
Retrieved Contexts
Top 1) Jon: The official opening night is tomorrow. …
⋮
Top 4) Jon: Tomorrow's gonna be an awesome night … I put so much into this and I want to savor all the good vibes.
Predicted Answer
Savor all the good vibes.

Figure 1: **Comparison between embedding-based and LLM-based retrieval.**

2 Related Work

2.1 Long Context Model Architectures

Handling long-context input is challenging due to the limitations of traditional transformer architectures (Vaswani, 2017), which scale quadratically with input length. Recent research has proposed various efforts to process longer sequences efficiently, while maintaining the model's ability to understand complex dependencies across tokens.

One common approach to addressing computational complexity involves modifying the attention mechanism. Models such as Longformer (Beltagy et al., 2020), BigBird (Zaheer et al., 2020), and Performer (Choromanski et al., 2020) have introduced more efficient alternatives to the traditional attention mechanism.

Another strategy focuses on altering the model architecture by incorporating recurrent structures, as seen in Transformer-XL (Dai, 2019) and Compressive Transformer (Rae et al., 2019). Hybrid models, such as Griffin (De et al., 2024), which combines local attention with recurrent blocks, and Reformer (Kitaev et al., 2020), which uses locality-sensitive hashing (LSH).

However, recent studies suggest that merely expanding the context length is not enough to improve model performance. Longer contexts can lead to the loss of relevant information, especially in the middle of the input (Liu et al., 2024). Moreover, the inclusion of unnecessary or irrelevant information in long contexts can significantly degrade model performance (Shi et al., 2023). Effectively utilizing long contexts remains a challenge, and this limitation persists in current models.

2.2 Retrieval-Augmented Generation

Retrieval-Augmented Generation (RAG) is a paradigm to enhance generative models by retrieving relevant information from external knowledge sources and using it as context (Lewis et al., 2020). This approach addresses issues in LLMs, such as hallucination and outdated knowledge, by grounding their outputs in external information.

Retrieval methods are broadly divided into two categories: sparse retrieval and dense retrieval. Sparse retrieval methods, like BM25 (Robertson et al., 1995) and TF-IDF (Sparck Jones, 1972), rely on exact term matching but often struggle with semantic relevance. Dense retrieval methods, such as DPR (Karpukhin et al., 2020), use embeddings to capture semantic similarity, outperforming sparse methods in various tasks including QA task.

Despite its advantages, RAG faces several challenges. As illustrated in Figure 1, embedding-based retrieval often prioritizes contexts containing exact query terms, overlooking logically relevant contexts without those terms. In contrast, LLMs can retrieve contexts that logically support the query even if they lack query terms. A detailed case study is provided in Appendix B.1.

Another challenge lies in setting the top-k, or the number of contexts to retrieve. Embedding-based retrievers require a fixed top-k, but the optimal number can vary by domains or chunk size. A large top-k may introduce noise, while a small top-k risks missing critical information. While similarity thresholds (Radeva et al., 2024) can dynamically adjust retrieval, they often require fine-tuning for individual data points.

To address these challenges, recent efforts incorporate query rewriting (Ye et al., 2023), adaptive search (Wang et al., 2023b; Jeong et al., 2024), verification (Li et al., 2023), and self-reflection (Asai

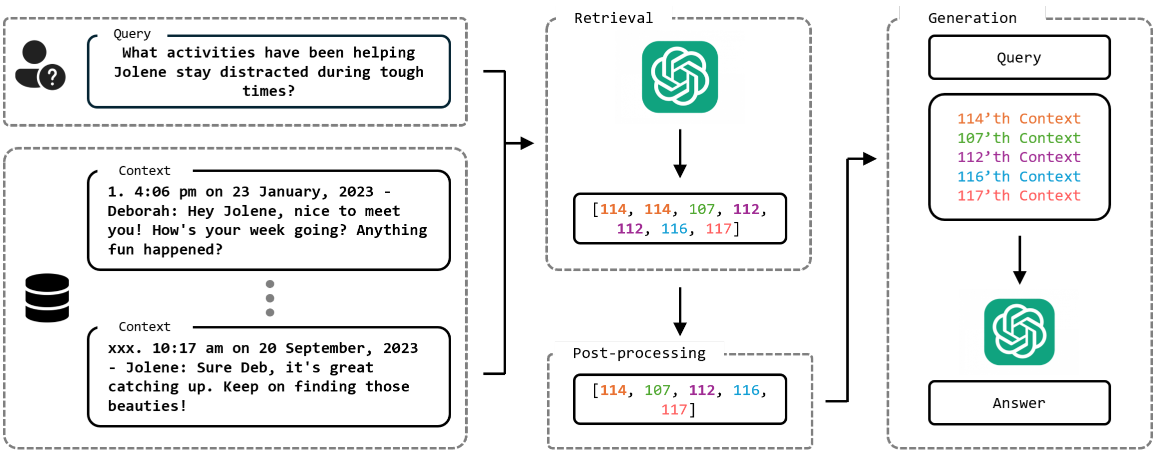

Query
What activities have been helping Jolene stay distracted during tough times?
Context
1. 4:06 pm on 23 January, 2023 - Deborah: Hey Jolene, nice to meet you! How's your week going? Anything fun happened?
Context
xxx. 10:17 am on 20 September, 2023 - Jolene: Sure Deb, it's great catching up. Keep on finding those beauties!
Retrieval
[114, 114, 107, 112, 112, 116, 117]
Post-processing
[114, 107, 112, 116, 117]
Generation
Query
114'th Context
107'th Context
112'th Context
116'th Context
117'th Context
Answer

Figure 2: **Framework of Letriever.** There are three stages. **1. Retrieval:** We let LLM retrieve indices of relevant contexts to the question. **2. Post-processing:** We remove the duplicates and preserve the order. **3. Generation:** We replace the indices with the original contexts. Finally, LLM answers to the query with the retrieved contexts.

et al., 2023; Li et al., 2024). We address these challenges by directly employing LLMs as retrievers, leveraging their advanced reasoning and contextual understanding capabilities

2.3 Prompt Engineering

Prompt engineering is a critical technique for optimizing the performance of large language models (LLMs) and is widely utilized to adjust model outputs across various tasks. Previous study (Brown et al., 2020) introduced Few-shot prompts with GPT-3, incorporating task examples into the input to enhance the model's contextual learning abilities. Subsequently, the impact of Few-shot prompt structure and expression on performance was analyzed, demonstrating that optimal prompt design can significantly improve model outputs (Scao and Rush, 2021). Additionally, the effects of prompt order and composition on model responses were evaluated, emphasizing the importance of fine-tuning prompt details (Perez et al., 2021).

The expanded application of prompt engineering for complex tasks has led to the development of Chain-of-Thought (CoT) prompts. CoT prompts have been shown to guide models to explicitly process intermediate reasoning steps, achieving superior performance in solving mathematical problems and logical reasoning tasks (Wei et al., 2022). Furthermore, CoT prompts have been effectively employed in Zero-shot settings, significantly enhancing the model's reasoning capabilities (Kojima et al., 2022). Furthermore, to refine multi-step reasoning across diverse tasks, combining CoT prompts with additional prompt instructions has also been proposed (Wang et al., 2022).

Prompt engineering also plays a pivotal role in domain-specific tasks. Customized prompts incorporating domain knowledge, such as in law or medicine, have been shown to significantly improve model accuracy (Mishra et al., 2022). Prompt designs reflecting specific vocabulary and writing styles have been found to outperform generic prompts in domain-specific tasks (Longpre et al., 2021). Additionally, a methodology for eliminating irrelevant information within prompts has been proposed, creating concise and effective designs that improve the efficiency of domain-specific applications (Min et al., 2022).

Our method also uitlizes prompt engineering techniques to summarize long input contexts.

3 Letriever

This outlines Letriever, which utilizes LLMs as retrievers in long-context question answering tasks. Our approach consists of three stages: Retrieval, Post-processing, and Generation. The overall framework is illustrated in Figure 2

3.1 Retrieval

We designed the retrieval phase to be as simple as possible to investigate the ability of LLM to retrieve contexts. We simply provided all the contexts C to the LLM and instructed it with instruction I_r to retrieve k most important contexts for answering the question q. As a result, LLM responds a Python list $L_{\text{raw}} = \{i_1, i_2, \ldots, i_n\}$ containing the indices of the retrieved context to ensure that the context is not modified. Note that the number of retrieved

```
Select {k} important contexts from
CONTEXT. Important contexts are those
that can help answer the QUESTION.
Provide the selected contexts'
positions (indices) in a Python
list. Provide only the indices as your
response. Assume that the index starts
from 0. The indices must also be less
than {context_len}. Output as a list.

## CONTEXT
{context}
## QUESTION
{question}
```

Figure 3: **Prompt for retrieval stage in Letriever framework.** For $k = no\_k$ setting, we simply do not provide $\{k\}$ to LLM, allowing the LLM to determine it dynamically.

indices n can be different from k. See Analysis 5.2 for more details. The process can be written as follows:

$$L_{\text{raw}} = LLM(I_r; C; q; k),$$

In contrast to traditional methods, this approach does not retrieve contexts based on embedding similarity with the query. Instead, it relies on the expectation that the LLM can identify relevant contexts that embedding similarity alone may fail to capture.

Like Figure 3, we provide the prompt used by the LLM to retrieve relevant contexts during the retrieval phase. Full contexts and a question are provided to the LLM. Then, the LLM is instructed to select the relevant contexts that can help answer the question. For the generation prompt, we adopt different prompts depending on the dataset.

3.2 Post-processing

After receiving the list of indices from the LLM, we further post-processed it. The LLM occasionally included duplicate indices in its response. We removed these duplicate indices. Additionally, the list of indices provided by the LLM was in no particular order; it was neither ascending nor descending. This suggests that the LLM arranged the indices based on their importance in answering the given question. We did not rearrange these indices; instead, we maintained the original order provided by the LLM. As a result, we get a final list of indices of relevant contexts L_p.

3.3 Generation

The generation phase is the same as the traditional embedding-based RAG method. We extract the retrieved contexts $C_{\text{retrieved}}$ by replacing the post-processed indices L_p with the original contexts. Given the retrieved contexts $C_{\text{retrieved}}$ and instruction I_g, a generation model finally generates an answer A based on the question q.

$$C_{\text{retrieved}} = \{C[l] \mid l \in L_p\},$$
$$A = LLM(I_g; C_{\text{retrieved}}; q)$$

4 Experiments

4.1 Datasets

LoCoMoQA (Maharana et al., 2024)**.** LoCoMo is a dataset of very long-term conversations with multi-sessions. We use 1,540 QA pairs in this dataset, excluding adversarial questions that a generation model should answer as ‘unanswerable’ because the evidence is absent, making them irrelevant to the performance of retrieval methods. These QA pairs include single-hop, multi-hop, temporal, and open-domain questions. In addition, we use two types of retrieval units in the dataset: dialogue and observation, the latter of which refers to information observed in the dialogue history (e.g., 'Caroline attended an LGBTQ support group recently and found the transgender stories inspiring').

QASPER (Dasigi et al., 2021)**.** QASPER is a dataset for question answering on scientific research papers. It consists of 5,049 questions over 1,585 Natural Language Processing papers. We conducted experiments using a dataset of 1,155 QA pairs, excluding unanswerable questions. Each question in the dataset was associated with multiple annotator-provided answers. To evaluate model performance, we calculated scores for all available answers and adopted the maximum score for each question as the final metric.

SQuAD 2.0 (Rajpurkar et al., 2018)**.** This dataset is designed to evaluate reading comprehension and question answering (QA) performance. It consists of multiple paragraphs per topic, each containing several question-answer pairs. Since individual paragraphs average around five sentences and do not provide long contexts, multiple paragraphs under the same topic were concatenated into a single context. One question-answer pair was extracted from each paragraph to create the QA dataset. The

Dataset	Retrieval Method	Answer Prediction		Evidence Retrieval		
		F1	ROUGE-L	Precision	Recall	F1
LoCoMoQA (Dialogue)	Full Context	39.1	38.5	0.3	**100.0**	0.6
	DRAGON	45.1	44.2	4.5	81.8	8.5
	$\text{E5}_{\text{mistral-7b}}$	45.2	44.3	2.4	87.1	4.7
	openai-embedding	47.3	46.6	10.3	77.6	18.2
	Letriever (Ours)	**48.7**	**48.0**	**46.1**	68.2	**55.0**
LoCoMoQA (Observation)	Full Context	28.7	27.7	0.5	**100.0**	1.0
	DRAGON	41.0	40.0	16.5	61.8	26.0
	$\text{E5}_{\text{mistral-7b}}$	40.7	39.7	9.1	65.8	16.0
	openai-embedding	41.8	40.9	17.0	63.2	26.8
	Letriever (Ours)	**42.9**	**41.9**	**41.4**	56.9	**47.9**
QASPER	Full Context	47.9	46.5	5.3	**100.0**	9.7
	DRAGON	42.9	41.2	13.6	88.2	21.9
	$\text{E5}_{\text{mistral-7b}}$	45.2	43.1	12.0	92.0	19.8
	openai-embedding	43.8	42.0	14.0	91.1	22.6
	Letriever (Ours)	**48.8**	**47.1**	**35.4**	76.6	**39.1**

Table 1: **Abstractive question answering performance on the LoCoMoQA and QASPER datasets**. The best performance is marked in **bold**. Results are based on F1-score, ROUGE-L metric for answer prediction, and precision, recall, and F1 scores for evidence retrieval performance.

Dataset	Retrieval Method	F1	EM
SQuAD	Full Context	63.0	31.7
	DRAGON	75.4	**44.6**
	$\text{E5}_{\text{mistral-7b}}$	74.2	44.3
	openai-embedding	74.9	41.7
	Letriever (Ours)	**75.5**	**44.6**

Table 2: **Extractive question answering performance on the SQuAD2.0 dataset**. The best performance is marked in **bold**. Results are based on F1-score and EM metric for answer prediction

dev dataset was used, resulting in 350 question-answer pairs for experimentation, excluding unanswerable questions.

4.2 Evaluation Metrics

Answer Prediction. We report F1 and ROUGE-L (Lin, 2004) scores for abstractive QA tasks, where the generative model is required to rephrase or summarize relevant information (e.g., LoCoMoQA and QASPER). For the extractive QA task, where the generative model is required to answer by identifying the specific span of text directly from the context (e.g., SQuAD), we report F1 and Exact Match (EM) scores.

Evidence Retrieval. Using the gold evidence labeled in the LoCoMoQA and QASPER datasets, we evaluate evidence retrieval performance based on Precision, Recall, and F1. High Recall indicates that the model successfully retrieves a large portion of the gold evidence context, while high Precision suggests that the retrieved context contains minimal noise. The F1 score provides a balance between these two metrics.

4.3 Experimental Setup

Baselines. We utilize two types of baseline retrieval methods: (1) **Full Context** inputs all contexts to a generation model. Their length is within token limit of the model. (2) **Embedding-based retrievers** retrieve relevant contexts from a vector database by calculating similarity scores between embedded contexts and the question. We employ *DRAGON* (Lin et al., 2023), $E5_{mistral\text{-}7b}$ (Wang et al., 2023a), and *openai-embedding* which is `text-embedding-3-large`[1].

Generation Model. To evaluate retrieval performance, we used a fixed generation model across all baselines and our proposed method. Specifically, we employed `gpt-4o-mini`[2], which supports inputs of up to 128K tokens.

Top-k. We evaluate the baselines and our method with top-k settings of 5, 10, 25, and 50. We also include a *no_k* setting, where the LLM dynamically

[1] https://platform.openai.com/docs/guides/embeddings

[2] https://platform.openai.com/docs/models#gpt-4o-mini

determines the number of contexts to retrieve. This setting is unavailable to embedding-based retrievers. Retrieval is performed at the sentence level.

4.4 Results and Discussions

Table 1 presents the results for abstractive QA, while Table 2 presents the results for extractive QA. For clarity, the results for each retrieval method in the table were obtained using the specific top-k setting that yielded the best answer prediction performance for that method and dataset. The detailed results with all top-k settings are presented in Appendix A.

As Table 1 and Table 2 show, Letriever achieves the best performance across both abstractive and extractive datasets, with F1 scores of 48.7 for LoCoMoQA(Dialogue), 42.9 for LoCoMoQA(Observation), 48.8 for QASPER, and 76.9 for SQuAD.

The Full Context method includes all required contexts, achieving 100% recall in retrieval accuracy. Despite containing all necessary evidence, it produced the poorest answer prediction performance, except on the QASPER dataset. This indicates that while LLMs can handle long contexts, their ability to utilize them effectively remains limited. Moreover, longer inputs often generate longer outputs, which can hinder performance in QA tasks requiring concise answers. A related case study is provided in Appendix B.2.

For the QASPER dataset, Table 1 shows that the Full Context method outperformed all baseline embedding-based retrievers but underperformed compared to our LLM-based retriever approach. These findings suggest that while RAG methods help reduce computational costs, they may fall short on certain datasets. In contrast, LLMs demonstrate strong potential for retrieving relevant contexts based on the query.

Regarding the evidence retrieval performance in Table 1, embedding-based retrievers consistently achieved higher Recall than our method. As discussed in Analysis 5.2, this is because LLM-based retrieval typically retrieves fewer contexts than the specified top-k. Since Recall reflects the proportion of evidence included in the retrieved content, retrieving more contexts generally leads to higher Recall. However, the answer prediction results reveal that higher Recall does not always translate to better performance. This highlights the importance of balancing sufficient retrieval of relevant content with minimizing noise in the process.

5 Analysis

5.1 Ablation Study

Method	Answer Prediction (F1)			
	Lcm (Dia.)	**Lcm (Obs.)**	**Sqd**	**Qasp**
Letriever (Ours)	37.8	**35.7**	77.8	42.8
w/ s.o.	36.1	33.6	76.3	43.0
w/ k.d.	**37.9**	**35.7**	**78.0**	42.5
w/ s.o. & k.d.	35.6	33.9	76.6	**43.2**

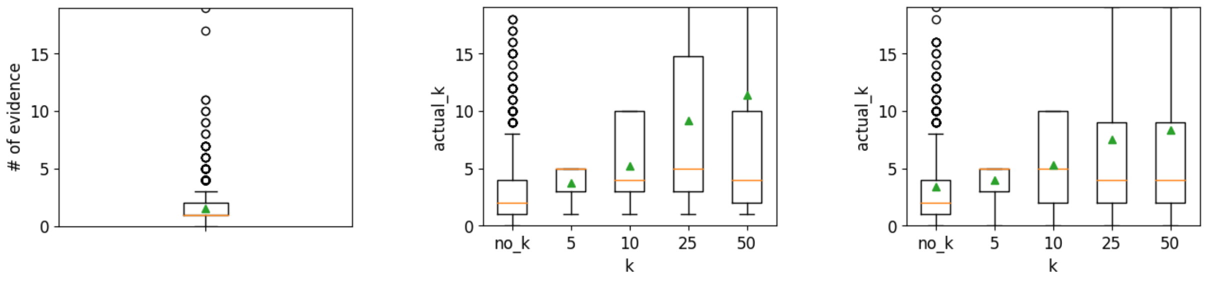
Table 3: **Ablation Study on Post-processing.** *s.o.* denotes sorting order, and *k.d.* denotes keeping duplicates.

Table 3 represents the answer prediction performance under post-processing ablation. To better assess the impact of each ablation on performance, samples where post-processing did not alter the results were excluded. Overall, except for QASPER dataset, preserving both order and duplicates achieved the optimal answer prediction scores. This indicates that LLM retrieves contexts in an order that results in better answer predictions and tends to duplicate important contexts. However, we removed duplicates in the main experiment because the baseline models cannot handle duplicates.

5.2 Analysis on Contexts Retrieved by LLM

We provide an analysis on the number of contexts retrieved by LLM in Table 4. Overall, the average number of retrieved contexts increases as the top-k increases. In addition, there is an interesting behavior on median value. In QASPER, the median value matched the top-k up to $k = 25$, but dropped sharply afterward. A similar pattern was observed in SQuAD, where the median matched the top-k up to $k = 10$, then dropped sharply and remained low.

This indicates that while the LLM attempts to follow the specified instructions, it tends to retrieve fewer contexts than the given number, relying on its own judgment. Notably, this judgment appears to align with improved answer prediction performance. For example, the overall median value for QASPER is significantly higher than that of other datasets, and its best performance was achieved at $k = 50$. In contrast, other datasets reached their best performance at $k = 5$ or $k = no_k$, highlighting the LLM's adaptability to different dataset requirements.



(a) Gold Evidences (b) Retrieved Dialogues (c) Retrieved Observations

Figure 4: **Comparison of distribution of the number of evidences on LoCoMoQA dataset**; **(a)**: the number of gold evidences, **(b)**: the number of evidences retrieved by LLM where the retrieval unit is dialogue, **(c)**: the number of evidences retrieved by LLM where the retrieval unit is observation. Among all of the k settings, no_k, which does not instruct the number of k, demonstrated the closest median and interquartile range.

Dataset	top-k	actual k (median)	actual k (avg.)	F1
LcmQA (Dia.)	5	5	3.7	**48.8**
	10	4	5.3	48.4
	25	5	9.1	47.1
	50	4	11.4	47.6
	no_k	2	28.7	48.7
LcmQA (Obs.)	5	5	4.0	42.8
	10	5	5.3	42.7
	25	4	7.5	42.5
	50	4	8.3	42.4
	no_k	2	3.4	**42.9**
QASPER	5	5	5.0	46.6
	10	10	9.3	48.4
	25	25	19.0	48.6
	50	10	35.6	**48.8**
	no_k	5	5.6	47.9
SQuAD	5	5	4.6	75.5
	10	10	7.3	75.1
	25	5	11.4	74.6
	50	5	11.8	74.5
	no_k	2	2.8	**77.4**

Table 4: **Analysis of the number of contexts actually retrieved by the LLM**, denoted as *actual k*, is provided. The numbers vary based on the top-k settings and datasets. We also present the answer prediction performance (F1) for each top-k setting to provide insight into the relationship between the top-k settings and answer prediction performance.

5.3 A Distributional Comparison with Gold Evidences

Figure 4 compares the distribution of gold-labeled evidences in the LoCoMoQA dataset with the evidences retrieved by the LLM. The gold evidence distribution suggests that the optimal number of evidences varies, even across questions within the same dataset. Traditional embedding-based retrievers, which use a fixed top-k setting, cannot replicate this variability, often leading to either the inclusion of noise or the omission of necessary evidence.

In contrast, LLMs can retrieve a variable number of contexts. While the number depends on the top-k setting, the *no_k* setting shows the closest median and interquartile range to the gold evidence distribution, whether the retrieval unit is dialogue or observation. Moreover, since LLM-based retrieval achieved the highest Precision and F1 scores, it strongly suggests that LLMs effectively minimize noise by dynamically selecting the appropriate number of contexts based on the query.

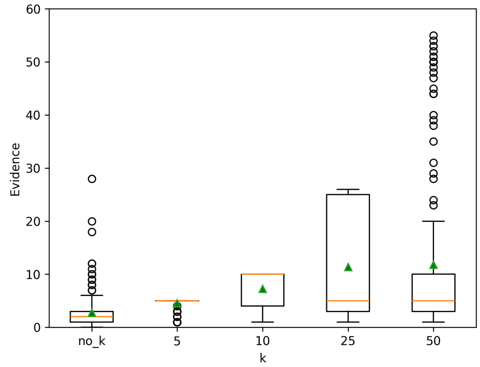


Figure 5: Distribution of the number of evidences in SQuAD 2.0

Figure 5 shows the distribution of the number of evidences in the SQuAD 2.0 dataset across various k settings. In SQuAD 2.0, the highest performance is achieved with the $k = 5$ setting, followed by the no-k setting. From the distribution table, we can observe that the evidence distribution for the no-k setting closely resembles that of the $k = 5$

setting. This suggests that when the LLM extracts evidence without being assigned a specific k value, it inherently selects the k value that it considers optimal for performance.

5.4 Performance Variation with Respect to Context Length

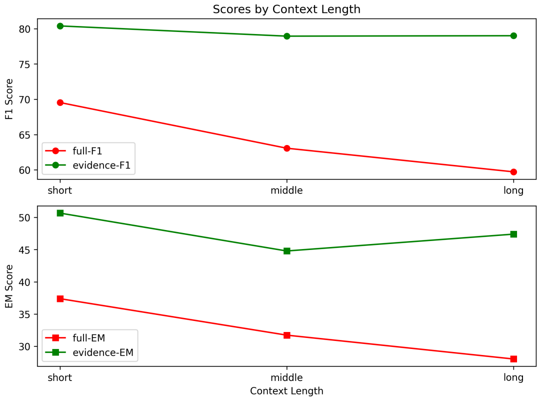


Figure 6: **Comparison of the performance variation between full context and Letriever (Ours) with different context lengths.** While the performance of full context decreases as the context length increases, Letriever (Ours) shows minimal performance variation.

Figure 6 illustrates the performance variation between full context and Letriever on SQuAD 2.0 under different context lengths. The "long" setting refers to the use of the full context without any modifications. The "middle" setting uses only the first two-thirds of the full context as input, while the "short" setting uses only the first one-third of the full context as input. In the SQuAD 2.0 dev set, there were not many datasets with contexts containing more than 300 sentences. Since this experiment only involves inference, we included the train set in the experiment as well. As the context length increases, the performance of full context gradually decreases, whereas Letriever (Ours) exhibits no change in performance. This demonstrates that as the context length grows, the LLM struggles to effectively process the full context. Consequently, the results indicate that summarizing the context enables the LLM to process it more effectively, as evidenced by the consistently high performance.

6 Conclusion

In this study, we propose Letriever, which leverages Large Language Models (LLMs) as contextual retrievers, and explored its potential for long-context Question Answering (QA) tasks. Our findings demonstrate that LLMs can effectively retrieve relevant information and adaptively process long contexts, resulting in higher answer prediction accuracy and retrieval precision compared to baseline methods. This robustness stems from the flexibility of varying top-k settings, which allows the model to retrieve an appropriate number of contexts based on different questions, and the ability to include contexts that do not contain words from the question but are logically essential.

Such adaptability enables LLM-based retrieval to address complex contextual nuances more effectively than traditional Retrieval-Augmented Generation (RAG) approaches that retrieve contexts based on embedding similarity.

7 Limitations

Despite the promising results demonstrated by Letriever, there are several limitations. First, it has primarily been evaluated on tasks within the context window limits of the LLM. We need to extend this methodology to address scenarios with contexts that exceed these limits, making it more scalable for real-world applications. Second, as discussed, LLMs struggle to effectively utilize long contexts, which could influence the performance of the retrieval stage in our approach, as the LLM needs to retrieve relevant contexts from the full context. A framework that reduces the number of contexts given in the retrieval stage may be necessary for higher performance. Therefore, our future work could focus on effectively addressing scenarios that were not discussed in this study and enhancing the retrieval performance of LLMs by reducing the length of the given contexts during the retrieval stage.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

AI Anthropic. 2024. The claude 3 model family: Opus, sonnet, haiku. *Claude-3 Model Card*, 1.

Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. 2023. Self-rag: Learning to retrieve, generate, and critique through self-reflection. *arXiv preprint arXiv:2310.11511*.

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Krzysztof Choromanski, Valerii Likhosherstov, David Dohan, Xingyou Song, Andreea Gane, Tamas Sarlos, Peter Hawkins, Jared Davis, Afroz Mohiuddin, Lukasz Kaiser, et al. 2020. Rethinking attention with performers. *arXiv preprint arXiv:2009.14794*.

Zihang Dai. 2019. Transformer-xl: Attentive language models beyond a fixed-length context. *arXiv preprint arXiv:1901.02860*.

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. *arXiv preprint arXiv:2105.03011*.

Soham De, Samuel L Smith, Anushan Fernando, Aleksandar Botev, George Cristian-Muraru, Albert Gu, Ruba Haroun, Leonard Berrada, Yutian Chen, Srivatsan Srinivasan, et al. 2024. Griffin: Mixing gated linear recurrences with local attention for efficient language models. *arXiv preprint arXiv:2402.19427*.

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*.

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*.

Nikita Kitaev, Łukasz Kaiser, and Anselm Levskaya. 2020. Reformer: The efficient transformer. *arXiv preprint arXiv:2001.04451*.

Takeshi Kojima, Shixiang Shane Gu, Machel Reid, Yutaka Matsuo, and Yusuke Iwasawa. 2022. Large language models are zero-shot reasoners. *Advances in neural information processing systems*, 35:22199–22213.

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.

Xiaonan Li, Changtai Zhu, Linyang Li, Zhangyue Yin, Tianxiang Sun, and Xipeng Qiu. 2023. Llatrieval: Llm-verified retrieval for verifiable generation. *arXiv preprint arXiv:2311.07838*.

Zhuowan Li, Cheng Li, Mingyang Zhang, Qiaozhu Mei, and Michael Bendersky. 2024. Retrieval augmented generation or long-context llms? a comprehensive study and hybrid approach. In *Proceedings of the 2024 Conference on Empirical Methods in Natural Language Processing: Industry Track*, pages 881–893.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Sheng-Chieh Lin, Akari Asai, Minghan Li, Barlas Oguz, Jimmy Lin, Yashar Mehdad, Wen-tau Yih, and Xilun Chen. 2023. How to train your dragon: Diverse augmentation towards generalizable dense retrieval. *arXiv preprint arXiv:2302.07452*.

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.

Shayne Longpre, Kartik Perisetla, Anthony Chen, Nikhil Ramesh, Chris DuBois, and Sameer Singh. 2021. Entity-based knowledge conflicts in question answering. *arXiv preprint arXiv:2109.05052*.

Junru Lu, Siyu An, Mingbao Lin, Gabriele Pergola, Yulan He, Di Yin, Xing Sun, and Yunsheng Wu. 2023. Memochat: Tuning llms to use memos for consistent long-range open-domain conversation. *arXiv preprint arXiv:2308.08239*.

Adyasha Maharana, Dong-Ho Lee, Sergey Tulyakov, Mohit Bansal, Francesco Barbieri, and Yuwei Fang. 2024. Evaluating very long-term conversational memory of llm agents. *arXiv preprint arXiv:2402.17753*.

Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2022. Rethinking the role of demonstrations: What makes in-context learning work? *arXiv preprint arXiv:2202.12837*.

Swaroop Mishra, Daniel Khashabi, Chitta Baral, Yejin Choi, and Hannaneh Hajishirzi. 2022. Reframing instructional prompts to GPTk's language. In *Findings of the Association for Computational Linguistics: ACL 2022*, pages 589–612, Dublin, Ireland. Association for Computational Linguistics.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation.

Ethan Perez, Douwe Kiela, and Kyunghyun Cho. 2021. True few-shot learning with language models. *Advances in neural information processing systems*, 34:11054–11070.

Irina Radeva, Ivan Popchev, and Miroslava Dimitrova. 2024. Similarity thresholds in retrieval-augmented generation. In *2024 IEEE 12th International Conference on Intelligent Systems (IS)*, pages 1–7. IEEE.

Jack W Rae, Anna Potapenko, Siddhant M Jayakumar, and Timothy P Lillicrap. 2019. Compressive transformers for long-range sequence modelling. *arXiv preprint arXiv:1911.05507*.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for squad. *arXiv preprint arXiv:1806.03822*.

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.

Teven Le Scao and Alexander M Rush. 2021. How many data points is a prompt worth? *arXiv preprint arXiv:2103.08493*.

Freda Shi, Xinyun Chen, Kanishka Misra, Nathan Scales, David Dohan, Ed H Chi, Nathanael Schärli, and Denny Zhou. 2023. Large language models can be easily distracted by irrelevant context. In *International Conference on Machine Learning*, pages 31210–31227. PMLR.

Karen Sparck Jones. 1972. A statistical interpretation of term specificity and its application in retrieval. *Journal of documentation*, 28(1):11–21.

A Vaswani. 2017. Attention is all you need. *Advances in Neural Information Processing Systems*.

Liang Wang, Nan Yang, Xiaolong Huang, Linjun Yang, Rangan Majumder, and Furu Wei. 2023a. Improving text embeddings with large language models. *arXiv preprint arXiv:2401.00368*.

Xuezhi Wang, Jason Wei, Dale Schuurmans, Quoc Le, Ed Chi, Sharan Narang, Aakanksha Chowdhery, and Denny Zhou. 2022. Self-consistency improves chain of thought reasoning in language models. *arXiv preprint arXiv:2203.11171*.

Yile Wang, Peng Li, Maosong Sun, and Yang Liu. 2023b. Self-knowledge guided retrieval augmentation for large language models. *arXiv preprint arXiv:2310.05002*.

Jason Wei, Xuezhi Wang, Dale Schuurmans, Maarten Bosma, Fei Xia, Ed Chi, Quoc V Le, Denny Zhou, et al. 2022. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837.

Feifan Wu, Lingyuan Liu, Wentao He, Ziqi Liu, Zhiqiang Zhang, Haofen Wang, and Meng Wang. 2024. Time-sensitve retrieval-augmented generation for question answering. In *Proceedings of the 33rd ACM International Conference on Information and Knowledge Management*, pages 2544–2553.

Fanghua Ye, Meng Fang, Shenghui Li, and Emine Yilmaz. 2023. Enhancing conversational search: Large language model-aided informative query rewriting. *arXiv preprint arXiv:2310.09716*.

Manzil Zaheer, Guru Guruganesh, Kumar Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, et al. 2020. Big bird: Transformers for longer sequences. *Advances in neural information processing systems*, 33:17283–17297.

Wanjun Zhong, Lianghong Guo, Qiqi Gao, He Ye, and Yanlin Wang. 2024. Memorybank: Enhancing large language models with long-term memory. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 38, pages 19724–19731.

A Detailed Experimental Results

The detailed question answering evaluation results with all top-k settings are presented in Table 5, Table 6 and Table 7.

Answer Prediction. The Full Context method showed the worst performance on LoCoMoQA and SQuAD, while in QASPER, all the RAG methods, except for Letriever (Ours), deteriorated compared to Full Context, regardless of *k*. This indicates limitations of RAG on certain datasets. LoCoMoQA and SQuAD achieved the best results when $k = no\_k$ or $k = 5$, whereas QASPER performed best with $k = 50$. This can be attributed to QASPER's need for a wider range of evidence. The performance improvement as the *k* value increases supports this trend. Additionally, QASPER differs from extractive datasets like SQuAD, where answers are concise. QASPER, in contrast, requires a deep understanding of the full context of scientific papers, necessitating the retrieval of a larger and more comprehensive set of relevant information.

Retrieval Accuracy. Regarding Retrieval Accuracy, Letriever performed the best when $k = 5$ or $k = 10$ in QASPER and slightly better than some baselines when $k = 5$ in LoCoMoQA. However, when *k* increases to 25 or 50, embedding-based retrievers achieve much higher Recall scores than Letriever. The rational for the result is discussed in Experiments 4.4. However, as discussed, a high Retrieval Recall does not necessarily correlate with better Answer Prediction performance. While it is important to include sufficient relevant content, it is equally crucial to reduce noise in the retrieval process to improve the Precision score. Regarding this, Letriever demonstrates its potential by achieving outstanding Precison and F1 scores, escpecailly when $k = no\_k$.

Retrieval Method	**Answer Prediction**	
	F1	**EM**
Full Context	63.0	31.7
	no_k	
Letriever (Ours)	**77.4**	44.3
	k = 5	
DRAGON	75.2	43.4
$\text{E5}_{\text{mistral-7b}}$	74.2	44.3
openai-embedding	74.9	41.7
Letriever (Ours)	75.5	**44.6**
	k = 10	
DRAGON	75.4	**44.6**
$\text{E5}_{\text{mistral-7b}}$	72.1	42.9
openai-embedding	72.3	41.1
Letriever (Ours)	75.1	43.1
	k = 25	
DRAGON	72.4	42.6
$\text{E5}_{\text{mistral-7b}}$	70.9	41.4
openai-embedding	70.2	39.4
Letriever (Ours)	74.6	43.4
	k = 50	
DRAGON	70.5	40.9
$\text{E5}_{\text{mistral-7b}}$	71.0	41.1
openai-embedding	69.3	38.6
Letriever (Ours)	74.5	43.1

Table 5: **Detailed question answering performance on SQuAD 2.0 dataset**. The optimal performance is marked in **bold**. Results are based on F1-score, EM metric for answer prediction; higher is better.

Retrieval Method	**Answer Prediction**				**Evidence Retrieval**					
	F1		**ROUGE-L**		**Recall@k**		**R-Prec**		**F1**	
	Dia.	Obs.	Dia.	Obs.	Dia.	Obs.	Dia.	Obs.	Dia.	Obs.
Full Context	39.1	28.7	38.5	27.7	**100.0**	**100.0**	0.3	0.5	0.6	1.0
no_k										
Letriever (Ours)	48.7	42.9	**48.0**	**41.9**	68.2	56.9	**46.1**	**41.4**	**55.0**	**47.9**
k=5										
DRAGON	43.2	41.0	42.5	40.0	63.7	61.8	16.1	16.5	25.7	26.0
E5$_{\text{mistral-7b}}$	43.9	39.8	43.1	39.0	62.7	59.7	15.5	15.6	24.9	24.6
openai-embedding	46.4	41.8	45.7	40.9	68.7	63.2	17.5	17.0	27.9	26.8
Letriever (Ours)	**48.8**	42.8	47.9	41.9	66.6	59.8	33.4	27.8	44.5	38.0
k=10										
DRAGON	44.4	40.7	43.6	39.8	72.7	66.4	9.6	9.3	17.0	16.3
E5$_{\text{mistral-7b}}$	44.8	40.7	43.9	39.7	71.5	65.8	9.2	9.1	16.3	16.0
openai-embedding	47.3	41.6	46.6	40.7	77.6	68.8	10.3	9.8	18.2	17.2
Letriever (Ours)	48.4	42.7	47.7	41.8	68.0	61.4	31.7	29.0	43.2	39.4
k=25										
DRAGON	45.1	39.2	44.2	38.2	81.8	71.9	4.5	4.3	8.5	8.1
E5$_{\text{mistral-7b}}$	44.8	40.4	43.9	39.4	81.3	72.0	4.4	4.3	8.3	8.1
openai-embedding	46.4	41.4	45.7	40.0	87.1	73.3	4.9	4.5	9.3	8.5
Letriever (Ours)	47.1	42.5	46.4	41.6	68.3	63.1	27.7	28.7	39.4	39.5
k=50										
DRAGON	44.7	38.5	43.7	37.5	87.5	74.7	2.5	2.3	4.9	4.5
E5$_{\text{mistral-7b}}$	45.2	39.1	44.3	38.1	87.1	75.3	2.4	2.3	4.7	4.5
openai-embedding	46.3	39.6	45.4	38.6	91.9	76.3	2.7	2.4	5.2	4.7
Letriever (Ours)	47.6	42.4	46.7	41.4	73.3	63.5	28.7	30.1	41.2	40.8

Table 6: **Detailed question answering performance on LoCoMoQA**. The best performance is marked in **bold**. Results are based on F1-score, ROUGE-L metric for answer prediction and Recall@k, R-Precision, F1 metric for evidence retrieval; higher is better.

Retrieval Method	**Answer Prediction**		**Evidence Retrieval**		
	F1	**ROUGE-L**	**Recall**	**Precision**	**F1**
Full Context	47.9	46.5	**100.0**	5.3	9.7
			no_k		
Letriever (Ours)	47.9	46.1	52.8	**57.9**	**47.6**
			k=5		
DRAGON	30.0	29.3	39.1	44.5	35.5
E5$_{\text{mistral-7b}}$	37.1	35.8	48.2	44.0	39.8
openai-embedding	37.2	36.0	43.0	49.2	39.0
Letriever (Ours)	46.6	44.7	51.9	52.9	46.0
			k=10		
DRAGON	36.4	35.2	56.8	35.0	37.5
E5$_{\text{mistral-7b}}$	40.5	38.8	65.0	33.1	38.5
openai-embedding	40.6	38.9	60.9	37.9	40.4
Letriever (Ours)	48.4	46.4	67.1	42.8	45.8
			k=25		
DRAGON	40.8	39.4	76.8	21.6	30.5
E5$_{\text{mistral-7b}}$	44.3	42.2	82.7	19.4	28.7
openai-embedding	43.3	41.8	80.9	23.1	32.5
Letriever (Ours)	48.6	46.8	76.7	35.6	41.2
			k=50		
DRAGON	42.9	41.2	88.2	13.6	21.9
E5$_{\text{mistral-7b}}$	45.2	43.1	92.0	12.0	19.8
openai-embedding	43.8	42.0	91.1	14.0	22.6
Letriever (Ours)	**48.8**	**47.1**	76.6	35.4	39.1

Table 7: **Detailed question answering performance on QASPER**. The best performance is marked in **bold**. Results are based on F1-score, ROUGE-L metric for answer prediction and Recall, Precision, F1 metric for evidence retrieval; higher is better.

B Case Study

B.1 Comparison between Embedding-based Retrieval and Our Method

Question: "What does Jon plan to do at the grand opening of his dance studio?"
Gold Answer: "savor all the good vibes"

Retrieval Method: DRAGON (top-k: 50)

> Retrieved Context List:
>
> Top-1. 8:29 pm on 13 June, 2023 - Jon said, "Thanks, Gina! It's been so inspiring to work with our young dancers, seeing their passion and commitment. Opening the dance studio's been a great experience - I want it to be a place of support and encouragement for all our dancers. Will you show me this presentation?"
>
> ...
>
> Top-50. 9:32 am on 8 February, 2023 - Jon said, "Thanks, Gina! Your pep-talk really meant a lot. I'm not gonna give up on my dreams - my dance studio and biz ventures need the hard work I'm putting in. Love having you in my corner, thanks for always being there!"

Recall@k: 0.0

Predicted Answer: "Let's make some awesome memories tomorrow at the grand opening!"

F1: 0.0

Retrieval Method: $\text{E5}_{\text{mistral-7b}}$ (top-k: 50)

> Retrieved Context List:
>
> Top-1. 10:04 am on 19 June, 2023 - Jon said, "Thanks, Gina. Still working on opening a dance studio."
>
> ...
>
> Top-50. 3:14 pm on 11 May, 2023 - Gina said, "It must be scary stepping into the unknown but I know you can do it, Jon. With your determination and drive, your dance studio will be a huge success. Keep that positive outlook and keep going!"

Recall@k: 0.0

Predicted Answer: "Let's make some awesome memories tomorrow at the grand opening!"

F1: 0.0

Retrieval Method: openai-embedding (top-k: 50)

> Retrieved Context List:
>
> Top-1. 4:04 pm on 20 January, 2023 - Jon said, "I've been into dancing since I was a kid and it's been my passion and escape. I wanna start a dance studio so I can teach others the joy that dancing brings me."
>
> ...
>
> Top-50. 10:04 am on 19 June, 2023 - Jon said, "Yeah! Let's make some awesome memories tomorrow at the grand opening!" and shared a photo of a man in a native costume is giving another man a high five

Recall@k: 0.0

Predicted Answer: "Can't wait for it - and for you to be there!"

F1: 0.0

Table 8: **A case study of embedding-based retrieval methods on LoCoMoQA.** All of the methods tended to prioritize contexts that include words in the given question, highlighted in red, but without a logical reason. The methods retrieved 50 contexts, but they failed to retrieve evidence that answers the question.

Question: "What does Jon plan to do at the grand opening of his dance studio?"
Gold Answer: "savor all the good vibes"

Retrieval Method: Letriever (Ours) (top-k: no_k)

> Retrieved Context List:
> Top-1. 10:04 am on 19 June, 2023 - Jon said, "The official opening night is tomorrow. I'm working hard to make everything just right. Can't wait to see it all come together!" and shared a photo of a group of young dancers in a dance studio
> Top-2. 10:04 am on 19 June, 2023 - Gina said, "Congrats, Jon! The studio looks amazing. You've put a lot of work into this and I'm so pumped for the launch tomorrow. Don't miss a beat!"
> Top-3. 10:04 am on 19 June, 2023 - Gina said, "Wow, Jon, you must be so excited! You've come so far since we last talked, and tomorrow's gonna be a blast! All those long nights were worth it - so take some time to savor it. Capture the joy and thrill that dance brings - it's magical!"
> **Top-4. 10:04 am on 19 June, 2023 - Jon said, "Tomorrow's gonna be an awesome night and I'm not gonna forget a second of it. I put so much into this and I want to savor all the good vibes. Thanks for always having my back. You're the best!"**
> Top-5. 10:04 am on 19 June, 2023 - Jon said, "Yeah! Let's make some awesome memories tomorrow at the grand opening!" and shared a photo of a man in a native costume is giving another man a high five

Recall@k: 100.0
Predicted Answer: "Savor all the good vibes."
F1: 100.0

Table 9: **A case study of the LLM-based retrieval method (Ours) on LoCoMoQA.** LLM was able to retrieve evidence that does not include words in the given question, highlighted in **bold**, but logically supports it. In this case, we did not specify the number of contexts to retrieve, and LLM retrieved 5 contexts, including that the grand opening is tomorrow and Jon plans to savor all the good vibes tomorrow.

B.2 Comparison between Full Context and Our Method

Example 1

Question: "What was the Norman religion?"
Gold Answer: "Catholicism"

Retrieval Method: Full Context
Predicted Answer: "The Normans replaced the Norse religion with Catholicism (Christianity)."
F1: 25.0

Retrieval Method: Letriever (Ours) (top-k: 5)

> Retrieved Context List:
> ...
> Top-2. Rollo's contingents who raided and ultimately settled Normandy and parts of the Atlantic coast included Danes, Norwegians, Norse–Gaels, Orkney Vikings, possibly Swedes, and Anglo-Danes from the English Danelaw under Norse control.The descendants of Rollo's Vikings and their Frankish wives would replace the Norse religion and Old Norse language with Catholicism (Christianity) and the Gallo-Romance language of the local people, blending their maternal Frankish heritage with Old Norse traditions and customs to synthesize a unique N̈ormanc̆ulture in the north of France.
> ...

Predicted Answer: "Catholicism (Christianity)"
F1: 66.7

Example 2

Question: "When did ARPNET and SITA become operational?"
Gold Answer: "1969"

Retrieval Method: Full Context
Predicted Answer: "ARPANET and SITA became operational in 1969."
F1: 28.6

Retrieval Method: Letriever (Ours) (top-k: 5)

> Retrieved Context List:
> Top-1. <Uyless Black, X.25 and Related Protocols, IEEE Computer Society, 1991> <Uyless Black, Frame Relay Networks, McGraw-Hill, 1998> <Uyless Black, MPLS and Label Switching Networks, Prentice Hall, 2001> < Uyless Black, ATM, Volume I, Prentice Hall, 1995>ARPANET and SITA HLN became operational in 1969.

Predicted Answer: "1969"
F1: 100.0

Table 10: **Case Study of the LLM-based retrieval method (Ours) on SQuAD 2.0.** When the full context is provided, the predictions tend to include the correct answer but are delivered in long sentences rather than concise responses. On the other hand, the Letriever (Ours) answers succinctly and provides the correct answer effectively.