Retrieval Augmented Generation (RAG) for Open Domain Complex Question Answering

Sebastiaan Beekman
5885116

Minouk Noordsij
4788338

Petar Petrov
5215781

Martijn Staal
4790162

Abstract

Question Answering (QA) is a well-studied natural language processing task, that seems to be a solved problem with the introduction of advanced Large Language Models. While these models seem to encode knowledge in their parameters, they are no perfect QA systems on their own. One approach to improve their performance is Retrieval Augmented Generation (RAG), in which the LLM is provided with extra context that a retriever has found based on the question, to improve QA performance.

In this research project, we have conducted experiments to evaluate what kind of context is valuable to provide the LLM with when answering compositional questions. What is the right mix of relevant, random negative and hard negative contexts? The results of our experiment point in the direction that relevant documents are necessary for good performance and that both hard negative and random negative documents deteriorate performance. However, due to the limitations of our research setup, we find that these results are not conclusive.

1 Introduction

Question Answering is a well-studied natural language processing task that can be applied in a wide variety of use cases. With the introduction of the more advanced Large Language Models (LLMs), easily deployable and correct Question Answering systems seem to be just within reach.

However, while it is believed that LLMs encode knowledge in their parameters, they are not impeccable question-answering systems out of the box (Lewis et al., 2020). For example, LLMs might seem to "come up" with false answers, known as hallucinations (Cuconasu et al., 2024). Furthermore, LLMs cannot be expected to correctly answer questions regarding recent events or developments that happened after the cutoff of their training data, and retraining LLMs regularly with newer data is expensive and wasteful of planetary resources due to the large amounts of energy required for LLM training (Wu et al., 2022).

Retrieval Augmented Generation (RAG) is a popular approach to still leverage LLMs for question-answering systems while limiting the aforementioned negative effects. It improves question-answering systems that rely on LLMs by providing the model with extra context using a pre-existing search engine (Cuconasu et al., 2024). In a RAG-based question-answering system, the extra context for the question is added to the LLM prompt using results from a search engine (Lewis et al., 2020). However, what kind of context should be given to the generator LLM to achieve the best results in a question-answering task remains an open question?

1.1 Research Questions

In this project, our goal is to study the impact of different types of context that are passed to the generator by the retriever in RAG systems for complex and open-domain question answering. In our research, we focus on the answering of compositional questions specifically. We distinguish three different types of contexts: negative, related and gold. Negative context can be subdivided into random and hard negative context. We use the following definitions for these types of contexts, combining concepts from two previous studies (Zhan et al., 2021; Cuconasu et al., 2024):

- *Gold context:* contains the answer and relevant information to the query.
- *Related context:* contains correct and relevant information that helps to answer the query. This means that gold context is also related context.
- *Hard negative context:* contains distracting information which is semantically similar to the query but does not contain the correct answer

or relevant information. It is usually scored high by the retriever despite not being related context.

- *Random negative context:* contains only information that is unrelated and irrelevant to answering the query. It can be obtained by randomly sampling from the set of all irrelevant documents in the corpus.

We investigate the impact of these different context types on question-answering performance in RAG systems by answering the following research questions:

1. How does using negative context impact downstream answer generation performance?
2. Is negative context more important for answer generation than related context?
3. Does providing only gold context deteriorate the performance compared to mixing with other negative or related contexts?

1.2 Structure

First, we discuss the theoretical background of our research in Section 2. We discuss related work of previous studies in Section 3. Then we discuss our research methodology in Section 4. We will discuss both the data used and our general experimental setup, as we discuss each experiment we have conducted for all three research questions.

In Section 5, we discuss the results of our experiments. In Section 6 we discuss the limitations of our research, and we conclude our report with our conclusions and possibilities for future work in Section 7.

2 Theoretical background

Question-answering is a category of information retrieval tasks with the common goal of providing the end-user with a complete answer to their question. Different types of question-answering tasks exist. A question-answering system might be focused on a particular domain such as medical or legal questions, but may also be a general system for questions from any domain ("open domain"). Furthermore, question-answering systems might be applied in different contexts, such as in a chatbot as an interactive question-answering system (Biancofiore et al., 2024).

Equally many approaches exist to implement question-answering systems. Recent developments in Large Language Models have introduced a new generative approach to question-answering systems where the initial question is the prompt for the LLM (Lewis et al., 2020), after being enriched by context gathered via a search engine. Such models are called Retrieval-Augmented Generation (RAG) models and consist of a retriever and a generator (Cuconasu et al., 2024). The retriever is a search engine which aims to find relevant documents to provide context with which the generator can produce an answer to the question. The context consists of documents retrieved by the retriever from a corpus. The generator is a pre-trained LLM. This setup combines parametric memory in the form of the pre-trained LLM with non-parametric memory in the form of the documents provided by the retriever.

3 Related work

Previous studies already investigated the impact of different types of context on the performance of RAG models for open-domain question answering. One of these studies employed hard negatives to optimize the retrieval step (Zhan et al., 2021). The approach Zhan et al. (2021) used is called dense retrieval (DR). It encodes the queries and the set of contexts into low-dimension embeddings to capture their semantic meanings, allowing the model to better distinguish between related and seemingly relevant but unrelated contexts.

Zhan et al. (2021) developed two training strategies: Stable Training Algorithm for dense Retrieval (STAR) and Algorithm for Directly Optimizing Ranking Performance (ADORE). STAR introduces random negatives and ADORE uses dynamic hard negative sampling to optimize the ranking performance. Experiments using retrieval benchmarks show that they both achieve significant performance improvements. The hard negative sampling strategy (ADORE) is better, but their combination achieves the best performance.

Another recent study investigated the impact of all different types of context and discovered 'the power of noise' (Cuconasu et al., 2024). The authors found that using random, noisy contexts increases the accuracy of RAG models if correctly positioned within a prompt. This is because the position of information within a prompt was found to be relevant as well. Relevant information should be placed near the query. In other cases, the attention scores showed worse performance. The best result is obtained when the gold context is near the query, the second best when the gold context is near the task instruction, and the worst when the gold document is placed in the middle of the context. The result also depends on the model, since different models exhibit distinct behaviors. However, in this study, we will not focus on document positions and will be randomly mixing them in the context.

Lastly, Cuconasu et al. (2024) found that top-ranked retrieved context that contain hard negative samples lower the effectiveness. They confirmed this using two different dense retrieval systems, Contriever and ADORE. They experimented with having 1, 2, and 4 distracting samples containing hard negative information and placed the golden information closest to the query. The authors found increasingly lower accuracies for these experiments and all results were significantly worse than the baseline accuracy where no distracting documents were included. Their results obtained with Contriever were better than that of ADORE.

4 Experiments

To answer our research questions, we have defined three experiments. In this section, we first discuss our general experimental setup used for all three experiments (Subsection 4.1). Then we discuss the data we have used (Subsection 4.2), after which we discuss the experimental setup of the baselines (Subsection 4.3) and all three experiments (Subsections 4.4, 4.5 and 4.6).

4.1 General experimental setup

For conducting our experiments we created Jupyter Notebooks based on the code used in previous research on this topic (Cuconasu et al., 2024). We have constructed a complete pipeline to run the required experiments, parts of which are inspired by or re-used from other papers, primarily the paper by Cuconasu et al. (2024). We use three different ways of obtaining contexts to pass to the generator of the RAG model for answering the query. We use Exact Match as a metric for evaluating generated answers.

In our experiments, we use two different retrieval setups. The first one is that we simply retrieve the gold contexts provided by the oracle. The second retrieval setup is based on Contriever. We use the pre-trained retriever Contriever to mine relevant documents (Izacard et al., 2022). These are the baseline experiments. Then, to see how different types of context impact downstream QA performance, we add different numbers of random or hard negative documents to the documents retrieved by the oracle or Contriever in the context provided to the LLM.

As our LLM, we use the 7B parameters version of the Llama2 family(Touvron et al., 2023). For all experiments we used the following prompt for the LLM: "You are given a question and you MUST respond by EXTRACTING the answer (max 5 tokens) from one of the provided documents. If none of the documents contain the answer, respond with NO-RES." followed by "\nDocuments:", the documents, "\nQuestion:", the question and "\nAnswer:"

The source code of our experiments is available on GitHub.[1]

4.2 Data

The data we use in our experiments are questions and the corpus from the 2WikiMultiHopQA dataset (Ho et al., 2020). The question data set contains compositional, comparison and inference-type questions. For each question, there are oracle contexts, supporting facts, evidence and answers. Our focus is on the compositional questions. We have limited our experiments to the first 1200 compositional questions from the *dev* subset, to keep evaluation feasible within the time set for this project.

4.3 Defining the baseline models

We start with defining the baseline models. The first one combines Contriever as the retriever with Llama2 as our LLM. We run Contriever with $k = 1$, $k = 3$, and $k = 5$ for retrieving top-k documents from the corpus.

As a second baseline, we repeat this experiment without the retriever, using only gold context from the provided in the dataset for each question. This means that we only use the oracle context that contains the necessary information to answer the question. For the compositional questions, there are, typically, two pieces of evidence incorporated in two different entries that contain the required information. The context is selected from the set of oracle documents if at least one of the provided pieces of evidence for this question is part of the context.

To illustrate this, if the question is *Where was the director of film Ronnie Rocket born?* and the

[1] https://github.com/SebastiaanBeekman/NLP-RAGS-QA/

evidence is `["Ronnie Rocket", "director", "David Lynch"]` and `["David Lynch", "place of birth", "Missoula, Montana"]`, we need the documents that contain *Ronnie Rocket* and *David Lynch* or *David Lynch* and *Missoula, Montana* to get to the answer: *Missoula, Montana*.

4.4 Determining the impact of negative contexts

To answer our first research question, we look at both sub-types of negative contexts: random and hard negative contexts.

To determine the impact of random negative context in our RAG setup, we randomly sample documents from the corpus and assume that they are not relevant to the current query. We combine the randomly sampled documents with the top-k relevant samples and use them as input to the LLM for answering a query. We experiment with adding 1 and 2 random documents to the top-k retrieved documents of Contriever using $k = 1$ and $k = 3$. We are forced to keep the total number of retrieved and sampled documents at 5 or lower due to the limitations in the dimension size of the chosen model.

Next, we will sample hard negatives from the oracle context for the current query instead of from random documents to determine their impact. We sample the documents that are part of the oracle context but do not contain the required evidence for a query. We decide which documents are the best hard negatives by measuring the dot product between the question and the context and taking the one with the largest value. Using the *nltk* library in Python, we tokenize the question and each non-golden oracle context example and we computed the dot product between them. Stop-words, as defined by the *nltk* library, were excluded to avoid having a large-valued dot product, which can be caused by unimportant words such as *the, and, a, no, he* & *she*. The resulting value represents the number of meaningful words that are in both the question and the context. For cases where there are too many documents with the same dot product value, we choose the shortest examples. We ensure this by adding the cosine similarity value, which is the dot product divided by the number of tokens in the context and question. This is a decimal value between 0 and 1. Similarly to the random samples, we also experiment with adding 1 and 2 hard negatives to the retrieved context.

Investigating the performance of these models will give us more insight into the impact of negative context.

4.5 Impact of negative context relative to related context

In answering our second research question, we use the results obtained from the experiment described before. Now, instead of focusing on the impact of negative context on its own, we will compare it to that of related contexts. We compare three different combinations of contexts:

1. The baseline which uses only documents retrieved by Contriever or from the oracle

2. A combination of retrieved documents with added random samples

3. A combination of retrieved documents where we added sampled hard negatives

This approach allows us to measure the impact that random and hard negative contexts have on question answering relative to the relevant context. In each prompt, all documents that are provided as context are listed in a random order so that the LLM cannot infer information from the order in which the documents are listed.

4.6 Comparing gold contexts to mixed contexts

Similar to the previous experiments, we aim to answer the third research question by running our setup several times using oracle context instead of Contriever retrieved documents. We described how to establish a baseline in Subsection 4.3 and how to sample random and hard negative context in Subsection 4.3. In this case, we do not have to set the parameter k to determine the number of retrieved entries, but we always retrieve all gold documents, meaning, we typically have two samples for each compositional question. Similarly to our setup with Contriever, we experiment with adding 1 and 2 random documents to the gold context, and with adding 1 and 2 hard negatives.

5 Results

The results of our experiments are summarized in Table 1 and discussed in this section. For each experiment, the first number is the absolute accuracy, and the second number is the relative accuracy. We define relative accuracy as how often questions are answered correctly when the retriever retrieves one or more contexts that contain the information required to answer the question correctly.

5.1 Baseline models

The performance of the Contriever and oracle baseline models on the 1200 compositional questions are listed in Table 1. It is interesting to note that the Contriever setup has, relatively, the best performance when it retrieves only one document because for answering compositional questions a combination of two pieces of evidence is necessary to be able to answer the question. The relative accuracy is the best accuracy achieved throughout this study. We suspect this is because it has the highest chance of making a correct guess when the answer is part of the only context the model can use for answering the question.

Overall, the absolute accuracy of our setup is quite poor when using Contriever. It seems to be the case that Contriever rarely succeeds in actually finding documents that contain the required evidence for the question. This could however be both a problem with the dataset we have used or with (our usage of) Contriever.

The oracle baseline has an accuracy of 0.26 despite having access to all golden context. When inspecting the generated answers, we found some possible reasons for this relatively low accuracy. Firstly, in some cases, the answer lacks some details, that are required to get an exact match. For example, the question *Where was the director of film Ronnie Rocket born?* was answered with *"Missoula"* instead of *"Missoula, Montana"*. Sometimes the answer was a bit more off, which makes it more questionable whether it should be right or wrong, for example, the question *Which country Aleksandra Marianna Wiesiołowska's father is from?* was wrongly answered with *"Poland"* instead of *"Polish-Lithuanian Commonwealth"*. There are also questions about which part of the evidence is not explicitly stated in any of the documents. For the question *Why did Charis Wilson's husband die?* there is a document about Charis Wilson being a subject of Edward Weston's photographs and about the cause of death of Edward Weston, but none of the documents contain the information that they were married. This is also the case for the question *Where does Huma Abedin's husband work at?*. We also saw this lack of specific information in the questions *What is the award that the performer of song Sunday Papers earned?*, where the documents only mentioned the performer being nominated for a Grammy award and *Where was the place of burial of Albert Frederick, Duke Of Prussia's mother?* where the document only mentioned the place of death. In these cases, the LLM responded with *"NO-RES"* as was requested in the prompt. Lastly, there are also cases where the LLM was provided the correct information but still answered wrongly. For example, the question *What nationality is the director of film Fúlmine?* where the documents stated the director was born in Spain and worked in Argentina, but the LLM responded with *"Argentine"*. We did not do an exhaustive study on the exact quantities of wrongly answered questions for each cause we mentioned above.

We also noticed the responses from the LLM were sometimes not adhering to the guidelines of the prompt. Instead of answering with solely the answer or *NO-RES*, the generated answers from all of our experiments often contained more than this. When generating an answer, it sometimes added an explanation on a new line and even more often added *NO-RES* on a new line. When checking the answers for correctness, we checked whether the correct answer was part of the generated answer, so the additions did not influence the accuracy. We did not encounter multiple answer attempts in any of the generated answers, so we judged that the LLM was not trying to "cheat". When the LLM was not able to answer, it sometimes generated strings like *"NO RESPONSE\nExplanation: None of the provided"*, *"UNKNOWN. None of the provided documents mention the birthplace of"* or *"Document [392398] - NO RESPON"*. These messy responses instead of adhering to the guidelines made it difficult to measure in how many of the cases the LLM was not able to answer.

5.2 Impact of negative contexts

From our results, we can see that using negative contexts deteriorates downstream answer generation performance for both random and hard negatives, both when these negative documents are added to documents retrieved by Contriever and when these are added to gold documents from the oracle.

5.3 Negative contexts versus relevant contexts

With the results of our experiments, we can also compare the impact of negative contexts with related contexts. We assume here that the documents retrieved by the Contriever are related, although we

Retrieval		Contriever k=1	Contriever k=3	Contriever k=5	Oracle
Baseline		0.0417 / 0.6667	0.03 / 0.1731	0.0284 / 0.1227	0.26 / 0.2624
Random Negatives	**1**	0.0225 / 0.3	0.0292 / 0.1628	-	0.1808 / 0.1825
	2	0.0033 / 0.1053	0.0008 / 0.0208	-	0.1258 / 0.127
Hard Negatives	**1**	0.0233 / 0.2772	0.0058 / 0.1029	-	0.1358 / 0.1371
	2	0.01 / 0.1538	0.0125 / 0.163	-	0.0967 / 0.0976

Table 1: Accuracy and relative accuracy respectively of top-k contexts for Contriever and Oracle baseline and models with 1 or 2 negative contexts added.

have reasons to believe that this is not always the case.

We see that adding negative context generally affects QA performance negatively. In all our experiments both absolute and relative accuracy is lower than the baseline when random negative or hard negative documents are added to the context.

This would suggest that it is mainly relevant context that is required for QA performance in an RAG-based QA system. However, due to the limitations in our experimental setup, we were not able to experiment with providing more documents or using few-shot learning so that the LLM "understands" what it can do with the negative context. We think that there might be a threshold of relevant context that must be provided before adding hard or random negative context can help increase QA performance, and that in our experiments this threshold was not met.

5.4 Gold contexts

Finally, we can compare the results of our experiments using gold context. From Table 1, we can see that, barring the relative performance for the baseline Contriever $k = 1$ setup, purely gold context outperforms every other experimental setup; this makes sense as gold context includes the necessary information to produce a correct answer. However, we should mention that performance is still not optimal, and the LLM we used only achieved an accuracy of 26%. As can be observed in the results of our Contriever baseline experiments, the LLM can get confused by supporting information that does not directly include the answer. We believe few-shot learning is a possible solution for this problem, but we did not test this due to time limitations. We can also observe that adding supplementary documents reduces the accuracy of the RAG system, with the worst performance coming from the setup where we add two hard negative examples. Intuitively, adding non-relevant documents to the context introduces unnecessary noise to the system, possibly confusing the LLM and hurting the overall accuracy. Interestingly enough, adding hard negatives results in the worst performance when using the oracle retriever, while in the Contriever-based experiments, the random documents result in the worst performance.

6 Limitations

In this research project, we have looked at the impact of different types of context provided in an RAG-based QA setup. However, we have evaluated our experiments using only one LLM and only one retriever. We expect that our results will be different when run on different models and using different retrievers. Future research should look at the effects of different types of contexts across multiple LLMs and retrievers. Also, due to limitations in available computing resources, we were not able to employ larger and more state-of-the-art LLMs, which might have performed generally better or reacted differently to the types of documents provided as context. This also limited the number of documents we could provide as context to the LLM. Providing more documents to the LLM could give different results and opens more possibilities for providing the LLM with different ratios of relevant, hard negative and random negative documents in the context.

Related to this is that we could not provide metrics on the effectiveness of our Contriever-based retrieval phase on the dataset we used, because we did not have complete information on which documents are deemed relevant to each question. Since the retrieval phase is an essential part of an RAG-based QA system, this is important information to have to provide a complete view of the performance of a complete system.

7 Conclusions

In conclusion, we find that in our experiments the better the context provided to the LLM, the better the QA performance. Adding random or hard negative documents to the context resulted in worse absolute and relative accuracy in all performed experiments.

Our experiments were however limited in several ways, namely only using one retriever and one LLM on just a small set of one dataset. Furthermore, we were limited in the number of documents we could provide in the context of the LLM. This limited the possibilities we had in providing different mixes of relevant, hard negative and random negative documents to the LLM. Lastly, our results are limited by the seemingly quite bad performance of Contriever in our experiments. Because of these limitations, we deem our results as inconclusive. The poor QA performance could, at least partially, be explained by a combination of these limitations.

7.1 Future work

As mentioned, in this research we focused on answering open-domain compositional questions. How well our results map to different types of questions could be researched further. Furthermore, the results might also be different if the questions are limited to a specific domain, which could also allow for domain-specific fine-tuning of the LLM used.

Lastly, few-shot learning is also a possibility to enhance LLM QA performance. Using few-shot learning might also increase the effectiveness of providing hard negative and random negative documents as context since it provides examples of what the LLM must do with such seemingly unrelated documents.

References

Giovanni Maria Biancofiore, Yashar Deldjoo, Tommaso Di Noia, Eugenio Di Sciascio, and Fedelucio Narducci. 2024. Interactive Question Answering Systems: Literature Review. *ACM Computing Surveys*, 56(9):239:1–239:38.

Florin Cuconasu, Giovanni Trappolini, Federico Siciliano, Simone Filice, Cesare Campagnano, Yoelle Maarek, Nicola Tonellotto, and Fabrizio Silvestri. 2024. The Power of Noise: Redefining Retrieval for RAG Systems. ArXiv:2401.14887 [cs].

Xanh Ho, Anh-Khoa Duong Nguyen, Saku Sugawara, and Akiko Aizawa. 2020. Constructing a multi-hop QA dataset for comprehensive evaluation of reasoning steps. In *Proceedings of the 28th International Conference on Computational Linguistics*, pages 6609–6625, Barcelona, Spain (Online). International Committee on Computational Linguistics.

Gautier Izacard, Mathilde Caron, Lucas Hosseini, Sebastian Riedel, Piotr Bojanowski, Armand Joulin, and Edouard Grave. 2022. Unsupervised Dense Information Retrieval with Contrastive Learning. ArXiv:2112.09118 [cs].

Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. 2020. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. In *Advances in Neural Information Processing Systems*, volume 33, pages 9459–9474. Curran Associates, Inc.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023. Llama 2: Open foundation and fine-tuned chat models.

Carole-Jean Wu, Ramya Raghavendra, Udit Gupta, Bilge Acun, Newsha Ardalani, Kiwan Maeng, Gloria Chang, Fiona Aga, Jinshi Huang, Charles Bai, Michael Gschwind, Anurag Gupta, Myle Ott, Anastasia Melnikov, Salvatore Candido, David Brooks, Geeta Chauhan, Benjamin Lee, Hsien-Hsin Lee, Bugra Akyildiz, Maximilian Balandat, Joe Spisak, Ravi Jain, Mike Rabbat, and Kim Hazelwood. 2022. Sustainable ai: Environmental implications, challenges and opportunities. In *Proceedings of Machine Learning and Systems*, volume 4, pages 795–813.

Jingtao Zhan, Jiaxin Mao, Yiqun Liu, Jiafeng Guo, Min Zhang, and Shaoping Ma. 2021. Optimizing Dense Retrieval Model Training with Hard Negatives. In *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, SIGIR '21, pages 1503–1512, New York, NY, USA. Association for Computing Machinery.