# RAGs for Open Domain Complex QA

## 1 Introduction

Question Answering is a well known task that has widespread applications in chatbots and domains that require precise information needs [10]. In real-world applications like healthcare, law and technical support, advanced information needs leads to more complex questions than simple factoid or extractive questions. Complexity in QA may arise from compositionality [11, 5] or requirement to gather evidence from multiple sources [2, 9] or numerical reasoning [8].

While there have been numerous supervised and unsupervised approaches for tackling complex QA, with the advances in Large Language Models [12, 1] the paradigm has shifted from training multiple models. With increase in scale, these models demonstrate emergent capabilities like In-Context Learning (ICL) [13]. ICL enables the model to learn the abilities required for the task from few demonstration samples through few-shot prompting.

These models are believed to encode world knowledge in their parameters. Hence, they are mostly evaluated in a closed book setting where the models are directly prompted to answer a given question, optionally given few demonstration samples using the parametric knowledge. However, this confines their ability to answer questions that require recent knowledge and also increases the probability of hallucinations [6] where the model generates factually incorrect responses. To combat such limitations and improve the performance of LLMs Retrieval Augmented Generation (RAG)[4, 7] is a promising solution. In RAG, the LLM is provided with contexts/snippets relevant to a query and the generation of answers is conditioned on the retrieved contexts than just relying on the pre-training knowledge encoded in LLM parameters. While the most relevant contexts are usually provided along with the query, more recent works [3] have observed the surprising effect of improved performance with the introduction of random documents. The authors hypothesize that introducing random documents increases the entropy of attention scores, which prevents entropy collapse, leading to an increase in performance. However, a detailed systematic study of variation in performance with variation in quality of contexts is not available.

In this project, our goal is to study the impact of contexts in RAG systems for complex QA and in particular our focus would be on compositional questions from 2WikiMultiHopQA.

## 2 Research Questions

Our goal is to answer the following research questions from this project

- **RQ1**:How does negative contexts impact downstream answer generation performance?
- **RQ2**: Are negative context more important for answer generation than related contexts?
- **RQ3**: Does providing only gold contexts deteriorate the performance compared to mixing with other negative or related contexts?

## 3 Experiments

In this section, a list of experiments are specified which would help answer the above listed research questions. In all experiments use first **1200 questions from dev.json** as test set.

- Download the dataset from https://drive.google.com/drive/folders/1qIZcNcU2wtiJNr3BUyX2GIUtnHEfbQDi?usp=sharing. The collection to use for retrieving contexts can be found at https://drive.google.com/drive/folders/1aQAfNLq6HBOw4_fVnKMBvKA6cXJGRTpH?usp=sharing. The code for loading data and perform off the shelf retrieval can be found at https://github.com/VenkteshV/DEXTER.
- Use off the shelf retriever like contriever and extract contexts for each query to be given as input to a generative model. Use Exact Match or cover Exact Match as metric for evaluating generated answers. Experiment with k=1,3,5 for retrieving top-k contexts and report the performance on generating answers.

- Repeat the above experiment without the retriever, using only oracle contexts as input. Oracle contexts are annotated documents provided for each question in dev.json.
- Now randomly sample documents from the collection that are not relevant to the current query during inference on the evaluation set. Combine these documents with the top-k relevant documents and use them as input to the LLM for answering a query. You can decide the ratios to mix the relevant and the random documents that serve as noise. Analyze the performance. Does injecting noise deteriorate or improve the final performance ?
- In this step, we will adopt a more principled approach to sample negative documents to be used as input to the RAG setup. Using a retrieval model, sample hard negatives from the collection for the current query instead of random documents to inject as noise. hard negatives are documents that are related and close to the query in the vector space but do not help answer the question. This can be sampled by retrieving documents not in the list of ground truth documents for a query as measure by dot product.
- Train a retrieval model using ADORE [14]. ADORE is optimized with hard negatives in a dense retrieval setup. Hence, it may be able to discern more relevant documents from large collections and lead to improved downstream answer generation performance. Using this retriever, retrieve relevant contexts followed by answer generation using LLMs. Compare it to the baseline performance of contriever based LLM QA mentioned in step 2 above.

In your report detail the intuition behind the performance changes and also qualitative analysis of few prominent examples. Also detail how the experiments answer the various research questions in results analysis. Detail in particular how sampling of noise affects the QA performance and to what extent the quality of contexts play a role.

## References

[1] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. D. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, A. Askell, S. Agarwal, A. Herbert-Voss, G. Krueger, T. Henighan, R. Child, A. Ramesh, D. Ziegler, J. Wu, C. Winter, C. Hesse, M. Chen, E. Sigler, M. Litwin, S. Gray, B. Chess, J. Clark, C. Berner, S. McCandlish, A. Radford, I. Sutskever, and D. Amodei, "Language models are few-shot learners," in *Advances in Neural Information Processing Systems*, H. Larochelle, M. Ranzato, R. Hadsell, M. Balcan, and H. Lin, Eds., vol. 33. Curran Associates, Inc., 2020, pp. 1877–1901. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2020/file/1457c0d6bfcb4967418bfb8ac142f64a-Paper.pdf

[2] Z. Chen, W. Chen, C. Smiley, S. Shah, I. Borova, D. Langdon, R. Moussa, M. Beane, T.-H. Huang, B. Routledge *et al.*, "Finqa: A dataset of numerical reasoning over financial data," *arXiv preprint arXiv:2109.00122*, 2021.

[3] F. Cuconasu, G. Trappolini, F. Siciliano, S. Filice, C. Campagnano, Y. Maarek, N. Tonellotto, and F. Silvestri, "The power of noise: Redefining retrieval for rag systems," 2024.

[4] Y. Gao, Y. Xiong, X. Gao, K. Jia, J. Pan, Y. Bi, Y. Dai, J. Sun, M. Wang, and H. Wang, "Retrieval-augmented generation for large language models: A survey," 2024.

[5] X. Ho, A.-K. Duong Nguyen, S. Sugawara, and A. Aizawa, "Constructing a multi-hop QA dataset for comprehensive evaluation of reasoning steps," in *Proceedings of the 28th International Conference on Computational Linguistics*. Barcelona, Spain (Online): International Committee on Computational Linguistics, Dec. 2020, pp. 6609–6625. [Online]. Available: https://aclanthology.org/2020.coling-main.580

[6] Z. Ji, N. Lee, R. Frieske, T. Yu, D. Su, Y. Xu, E. Ishii, Y. J. Bang, A. Madotto, and P. Fung, "Survey of hallucination in natural language generation," *ACM Computing Surveys*, vol. 55, no. 12, pp. 1–38, mar 2023. [Online]. Available: https://doi.org/10.1145%2F3571730

[7] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W. tau Yih, T. Rocktäschel, S. Riedel, and D. Kiela, "Retrieval-augmented generation for knowledge-intensive nlp tasks," 2021.

[8] W. Ling, D. Yogatama, C. Dyer, and P. Blunsom, "Program induction by rationale generation: Learning to solve and explain algebraic word problems," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Vancouver, Canada: Association for Computational Linguistics, Jul. 2017, pp. 158–167. [Online]. Available: https://aclanthology.org/P17-1015

[9] P. Lu, L. Qiu, K.-W. Chang, Y. N. Wu, S.-C. Zhu, T. Rajpurohit, P. Clark, and A. Kalyan, "Dynamic prompt learning via policy gradient for semi-structured mathematical reasoning," 2023.

[10] R. S. Roy and A. Anand, "Question answering for the curated web: Tasks and methods in qa over knowledge bases and text collections," 2022.

[11] H. Trivedi, N. Balasubramanian, T. Khot, and A. Sabharwal, "MuSiQue: Multihop questions via single-hop question composition," *Transactions of the Association for Computational Linguistics*, vol. 10, pp. 539–554, 2022. [Online]. Available: https://aclanthology.org/2022.tacl-1.31

[12] J. Wei, X. Wang, D. Schuurmans, M. Bosma, B. Ichter, F. Xia, E. Chi, Q. Le, and D. Zhou, "Chain-of-thought prompting elicits reasoning in large language models," 2023.

[13] S. M. Xie, A. Raghunathan, P. Liang, and T. Ma, "An explanation of in-context learning as implicit bayesian inference," 2022.

[14] J. Zhan, J. Mao, Y. Liu, J. Guo, M. Zhang, and S. Ma, "Optimizing dense retrieval model training with hard negatives," 2021.