The Impact of Embedding Methods on Faithfulness of Retrieval-Augmented Generation (RAG)

Amandeep Kaur Singh
amandeepkaur@umass.edu
University of Massachusetts
Amherst, USA

Chang Jin
changjin@umass.edu
University of Massachusetts
Amherst, USA

Reagan Keeney
rkeeney@umass.edu
University of Massachusetts
Amherst, USA

ACM Reference Format:
Amandeep Kaur Singh, Chang Jin, and Reagan Keeney. 2024. The Impact of Embedding Methods on Faithfulness of Retrieval-Augmented Generation (RAG). In *COMPSCI 646: Advanced Information Retrieval, Fall 2024, Amherst MA.* ACM, New York, NY, USA, 3 pages.

1 Problem Statement

Language models are a study topic with a long history, dating as early as 1950. Since then, language models have been applied in dynamic tasks such as information retrieval, speech recognition, and more. Large Language Models (LLMs) refer to language models that are trained on vast volumes of data, making them capable of answering questions in broad contexts and offering impressive natural language generative capabilities [11]. LLMs are designed to generate original output for tasks such as answering questions, translating between languages, and completing sentences. Despite their capabilities, LLMs are not without flaws, due to their text-generative nature and their need to answer all queries, LLMs are vulnerable to 'hallucinations' [10], where the model incorrectly answers questions with false information instead of reporting a lack of adequate topic knowledge. Such hallucinations can be dangerous as they may propagate misinformation, and lead to poor decision-making in critical applications such as healthcare, law, and more.

Retrieval-augmented generation (RAG) is a technique that optimizes a large language model's output, utilizing information retrieval to consult documents from a reliable knowledge base outside of its training data sources before generating a response using an LLM. Ensuring these LLM-generated responses are accurate, pertinent, and helpful is important. Current research faces challenges in assuring the faithfulness of the generated responses—that is, that the output stays correct and in line with the acquired data—a significant problem with RAG[5].

The selection of embedding techniques, which translate textual data into numerical vectors used to compute similarities and retrieve pertinent documents, is a key determinant of a Retrieval-Augmented Generation model's faithfulness and performance. Different embedding strategies can lead to drastically different retrieval accuracy, quality, and reliability of the generated responses. For instance, dense embeddings like BERT can capture semantic relationships, improving relevance and potentially maintaining faithfulness [2]; sparse embeddings, on the other hand, yield faster results but may sacrifice contexts, which can lead to less faithful responses [8].

Given these variations in embedding models, optimizing RAG for applications that need high factual consistency and alignment is important. Our study thus attempts to answer the following question: **Which embedding technique best maintains faithfulness in responses produced by RAG models on a domain-specific dataset?** We hope that our findings will support applying RAG for high-stake applications where maintaining faithfulness and building user trust (through interpretability) is crucial.

2 Motivation

Retrieval-augmented generation (RAG) has shown significant promise in enhancing the performance of large language models with the integration of retrieved information. The performance of RAG models can vary depending on the choice of embedding models used for information retrieval.

As such, while retrieval-augmented generation (RAG) offers a solution to some of the major shortcomings of Large Language Models (LLMs) for question-answering tasks, optimizing this pairing of systems remains inadequately explored. By systematically evaluating and comparing the faithfulness of RAG with different embedding models for a domain-specific dataset, we hope to gain valuable insights into how the choice of embeddings impacts the quality of the generated responses.

3 Related Work

Retrieval-augmented Generation for Knowledge-Intensive NLP Tasks' by Lewis et. al[10] introduced Retrieval-Augmented Generation as a novel approach combining information retrieval (IR) with text generation for knowledge-intensive Natural Language Processing (NLP) tasks in 2020. The proposed approach includes two types of memories:

1. Non-parametric Memory: Document index which can be 'hot-swapped' in order to update knowledge as needed.
2. Parametric Memory: Parameters of a generative model such as BART.

A key finding of this paper was that the proposed RAG models perform better and tend to be more 'grounded' i.e., they generate "more specific, diverse and factual language" than the then state-of-the-art parametric-only seq2seq baseline chosen for the study (BART). Lewis et. al concluded their findings with a discussion of the broader impact of their work, suggesting applications for medical information retrieval. This is particularly relevant for our

Conference'17, July 2017, Washington, DC, USA

ACM ISBN 978-x-xxxx-xxxx-x/YY/MM

proposed project, where we plan to evaluate the impact of the choice of embedding models on the performance of RAG by applying it to the RAG-Mini-BioASQ dataset.

More recently, Şakar and Emekci[15] performed 23,625 grid-search iterations to obtain results comparing several RAG architectures and techniques. The embedding models compared included OpenAI's text-embedding-v3-large model, Beijing Academy of Artificial Intelligence's (BAAI) open-source bge-en-small model, and Cohere's cohere-en-v3 model. As discussed in the results section of [15], the choice of embedding models significantly affects the performance of the RAG systems in terms of hardware utilization (runtime, CPU usage, and memory usage), and performance (evaluated with median similarity scores). While Şakar and Emekci's research included a direct comparison of embedding models, the focus was on efficiency with regard to runtime and hardware requirements, not on the quality of generated outputs. Which embedding model is most optimal seems to vary by use case [4, p.9], leaving ambiguity on the correct conditions for a given model.

As such, evaluating the retrieval quality and the generated output of RAG models remains a challenging task. Evaluating whether a model's outputs contain the correct information requires specific evaluation metrics and strategies. [4] explores some of the challenges in evaluating RAG systems and proposes 'A Unified Evaluation Process of RAG (AUEPORA) as a structured approach to RAG evaluation.

For this research, we focus on 'faithfulness' as a measure of generated output quality. Broadly, faithfulness is the level of agreement between the generated text and the contexts of all claims[1]. Broadly, faithfulness is a metric commonly used to understand to what extent the LLM hallucinates when generating responses[7].

4 Approach

4.1 Baseline

To determine the set of embedding models for testing, we referred to Muennighoff et al.'s *Massive Text Embedding Benchmark* (METB) [12] and its corresponding leaderboard.

As our baseline, we selected the Beijing Academy of Artificial Intelligence's open source model BGE-SMALL-EN-V1.5[14], which is the current version of the top performing model in Şakar and Emekci's resource efficiency comparison [15]. This model serves as a benchmark for evaluating the performance of other models in our study.

4.2 Models

We selected two additional models based on their rankings and parameter size relative to the baseline, using the METB leaderboard as a reference [12].

4.2.1 Baseline: BAAI/bge-small-en-v1.5 (rank 70 on MTEB). As introduced earlier, this was the Beijing Academy of Artificial Intelligence's open model, a top-performing model in Şakar and Emekci's resource efficiency comparison [15]. This model ranks 70 in the English Retrieval section of the METB and serves as our baseline for evaluating embedding techniques.

4.2.2 Model 2: abhinand/MedEmbed-small-v0.1(rank 58 on MTEB). MedEmbed is a model fine-tuned for medicinal and clinical data, intended for use in medical and clinical contexts. We selected this model to investigate whether its domain-specific design will improve the faithfulness of the responses generated. Its size is comparable to the baseline model but ranks higher in the English Retrieval section of METB.

4.2.3 Model 3: sentence-transformer/all-MiniLM-L6-v2(rank 133 on MTEB). This model is a sentence-transformer model that performs well with semantic search, clustering, and sentence similarity [16]. This model presents an advantage due to its' small size, a bonus that its size is similar to at of the baseline model. In the English Retrieval section of METB, this model ranks lower than our baseline at 133.

5 Experiments

5.1 Dataset(s)

For the purpose of our study, we plan to use a subset of the BioASQ 11b dataset [13] in the domain of biomedical question answering. Specifically, the pre-generated RAG-Mini-BioASQ dataset[3], which is designed for Retrieval-Augmented Generation use cases.

5.1.1 Key Features of the RAG-Mini-BioASQ dataset:

(1) Subset of BioASQ dataset: Given the limited resources and timeline of the project, using a smaller dataset is advantageous for ensuring the scope is achievable while the information ne still retain the complexity from the original dataset.
(2) Structured for RAG-based approaches/experiments: The RAG-Mini-BioASQ dataset already has a correct response to the question, which could be useful for evaluation purposes. The dataset consists of the following:
1. Text-Corpus: Consists of biomedical 40181 text passages and their IDs.
2. Question-Answer-Passages: Consists of 4719 questions (representing the users' information need), a correct answer to those questions, and a list of relevant passage IDs from the Text-Corpus data.
(3) Biomedical focus: The chosen dataset is domain-specific; enabling us to perform a more accurate assessment of various embedding models for this specific use case.
(4) Complexity: The questions are designed to be particularly challenging[9], which may make disparities between the faithfulness of different embeddings more apparent.

In particular, in critical domains such as biomedical question-answering, hallucinations from generative language models can raise significant safety concerns. Thus it is uniquely important to understand, evaluate, and ensure faithfulness to verify correct reference information in such fields.

5.1.2 Data Considerations: Given that the chosen dataset is already processed and designed for RAG tasks, it does not require extensive additional pre-processing. Additionally, due to computing limitations, we might randomly sample a subset of the queries and evaluate the RAG with the different embedding models on those queries only.

5.2 Overview of RAG Pipeline

Our Retrieval Augmented Generation pipeline begins by embedding passages using one of three models:

(1) Baseline: bge-small-en-v1.5,
(2) Fine-tuned BERT Model MedEmbed-small-v0.1,
(3) or all-MiniLM-L6-v2

The embeddings are stored in the Milvus vector database. When a query or question (information need) is passed in, it is first embedded, and then we retrieve documents from the RAG-Mini-BioASQ corpus using Milvus, leveraging vector similarity search. Then, we construct a prompt with the retrieved passages as context, and finally, Mistral AI's Mixtral-8x7B-Instruct-v0.1 Large Language Model generates natural language answers to the prompt based on the retrieved documents. Referenced documentation[18] to build the pipeline.

Detailed overview of the pipeline:

(1) Embedding the passages in the corpus (documents to retrieve from) using one of the three embedding models.
(2) Save the embeddings in the Milvus Vector Database: Milvus is a high-performing, open-source vector database optimized for efficient/fast, and scalable similarity searches on high-dimensional data [11, 17, 18]. It enables efficient retrieval of the most relevant documents for each query.
(3) Embedding the query and retrieving: For our experiments so far, we return the top 3 closest documents using the inner product as the similarity metric.
(4) Prompt Construction for LLM: Using the original question and the 3 retrieved documents as context, the prompt for the LLM model is formulated/structured. This step is crucial for ensuring that the generated answer is grounded in the retrieved evidence.
(5) Mistral AI LLM: For the final step of the RAG pipeline, we utilize MistralAI's Mixtral-8x7B-Instruct-v0.1 language model[6, 18]. This model was selected due to its capability to handle long prompts effectively and generate coherent, contextually aware responses, making it well-suited for our retrieval-augmented generation (RAG) setup.

5.3 Evaluation

Given that our research centers on the quality of the generated text, so do our evaluation metrics. We aim to use the RAG-specific evaluation of faithfulness, as defined in the DeepEval framework [1]. This metric aims to gauge how successfully the LLM incorporates context documents into its generated answer. To measure this, another "evaluator" LLM is utilized to identify claims made within both the answer and context. From there, the number of truthful(according to context passages) claims is divided by the total number of claims within the response. We intend to use faithfulness as a metric to monitor the degree to which the LLM integrates niche knowledge from the corpus to produce correct answers. A high faithfulness score would represent both integration and a lack of hallucination, both of which are important to ensure trustworthy models. Our specific tool for measuring faithfulness is from DeepEval's package, and employs a 4-bit quantized mistral-7b-v0.3 as the evaluator LLM. The RAG model output is provided with the retrieved context passages for evaluation.

References

[1] [SW] Confident AI, DeepEval version 1.4.7, Nov. 21, 2024. url: https://docs.confident-ai.com/.
[2] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*. Jill Burstein, Christy Doran, and Thamar Solorio, (Eds.) Association for Computational Linguistics, Minneapolis, Minnesota, (June 2019), 4171–4186. doi: 10.18653/v1/N19-1423.
[3] enelpol. 2024. Rag-mini-bioasq. (2024). https://huggingface.co/datasets/enelpol/rag-mini-bioasq.
[4] Yunfan Gao et al. 2023. Retrieval-augmented generation for large language models: a survey. *arXiv preprint arXiv:2312.10997*.
[5] Or Honovich et al. 2022. True: re-evaluating factual consistency evaluation. (2022). https://arxiv.org/abs/2204.04991 arXiv: 2204.04991 [cs.CL].
[6] Albert Q Jiang et al. 2023. Mistral 7b. *arXiv preprint arXiv:2310.06825*.
[7] Xiaonan Jing, Srinivas Billa, and Danny Godbout. 2024. On a scale from 1 to 5: quantifying hallucination in faithfulness evaluation. *arXiv preprint arXiv:2410.12222*.
[8] Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Bonnie Webber, Trevor Cohn, Yulan He, and Yang Liu, (Eds.) Association for Computational Linguistics, Online, (Nov. 2020), 6769–6781. doi: 10.18653/v1/2020.emnlp-main.550.
[9] Anastasia Krithara, Anastasios Nentidis, Konstantinos Bougiatiotis, and Georgios Paliouras. 2023. Bioasq-qa: a manually curated corpus for biomedical question answering. *Scientific Data*, 10, 1, 170.
[10] Patrick Lewis et al. 2020. Retrieval-augmented generation for knowledge-intensive nlp tasks. In *Proceedings of the 34th International Conference on Neural Information Processing Systems* (NIPS '20) Article 793. Curran Associates Inc., Vancouver, BC, Canada, 16 pages. isbn: 9781713829546.
[11] Shervin Minaee, Tomas Mikolov, Narjes Nikzad, Meysam Chenaghlu, Richard Socher, Xavier Amatriain, and Jianfeng Gao. 2024. Large language models: a survey. *arXiv preprint arXiv:2402.06196*.
[12] Niklas Muennighoff, Nouamane Tazi, Loïc Magne, and Nils Reimers. 2023. Mteb: massive text embedding benchmark. (2023). https://arxiv.org/abs/2210.07316 arXiv: 2210.07316 [cs.CL].
[13] n.d. 2023. Bioasq training 11b. (2023). http://participants-area.bioasq.org/datasets/.
[14] Beijing Academy of Artificial Intelligence. [n. d.] Bge-small-en-v1.5. (). https://huggingface.co/BAAI/bge-small-en-v1.5.
[15] Tolga Şakar and Hakan Emekci. 2024. Maximizing rag efficiency: a comparative analysis of rag methods. *Natural Language Processing*, 1–25.
[16] Rahultiwari Tiwari. 2024. Unlocking the power of sentence embeddings with all-minilm-l6-v2. (2024). https://medium.com/@rahultiwari065/unlocking-the-power-of-sentence-embeddings-with-all-minilm-l6-v2-7d6589a5f0aa.
[17] Jianguo Wang et al. 2021. Milvus: a purpose-built vector data management system. In *Proceedings of the 2021 International Conference on Management of Data*, 2614–2627.
[18] Cheney Zhang. 2024. Build rag with milvus. (2024). https://github.com/milvus-io/bootcamp/blob/master/bootcamp/tutorials/quickstart/build_RAG_with_milvus.ipynb.

6 Miscellaneous

[A] RAG pipeline as described in 5.2 **Click here** or go to https://colab.research.google.com/drive/1-Gg36PHDtDE3OT9WSd3zAX-AWMJCF_-x?usp=sharing
[B]Ongoing evaluation progress, refer to the second half of the document. **Click here** or go to https://colab.research.google.com/drive/11xlzUwWC2TkLH8k2KCSPTHDFVJ6aD2mt?usp=sharing
[C] Preliminary Data **Click here** or go to https://drive.google.com/file/d/1_coqwkR54wg-7o3qGhQpQ12mnm62tghi/view?usp=sharing