Retriever-Augmented Generation for Human Reference Atlas

Group 5: Rebecca Myers, Marissa Reed, Courtney Shammas
Github Repository: https://github.com/paytonncourt96/Deep_Learning

Introduction

The field of machine learning has consistently delivered advancements in the medical field. Retriever-Augmented Generation (RAG) is one such advancement. Pretrained large language models (LLMs) can store a certain amount of knowledge in the parameters and have shown good results when fine-tuning is applied to various downstream natural language processing (NLP) tasks. However, these LLMs need help manipulating and understanding domain-specific knowledge [6]. Given that these pre-trained models are developed with parametric memory, they cannot expand this memory and sometimes have "hallucinations." When we introduce non-parametric memory, i.e., retrieval-based, it becomes possible to revise and expand knowledge.

Enter RAG. RAG combines the capabilities of LLMs with external knowledge retrieval, enabling models to generate informed and contextually relevant outputs. RAG works as follows: A database contains documents or various pieces of data. Retrieval is conducted on the documents from the database in response to a query. The retrieved information is used to generate relevant output to the context. What is retrieved is based upon a prompt—this is considered augmenting the user's input query.

One compelling application of RAG is analyzing data from the Human Reference Atlas (HRA) [8]. The HRA goal is to take all the cells in the human body and map them to further research in the biomedical field. The complexity and heterogeneity of HRA data present challenges for traditional LLMs due to the need for domain understanding. When we use a targeted, domain-specific dataset and perform RAG, we are able to output responses to queries without fear of hallucination.

In this work, we develop a prototype Q&A system to assist users in accessing HRA. By employing RAG on HRA data, we can dynamically retrieve relevant biomolecular information. The collaborative approach of combining machine learning, and domain specific knowledge, provides more interpretable outputs without fear of hallucination, fostering a sense of shared progress in the research community.

Related Work

RAG has emerged as a powerful approach in NLP when it comes to tasks that require domain-specific knowledge. Circa 2021 saw the development of models like GPT-3 [9], BERT [10], and RoBERTa [11]. These widely used models can be credited to the introduction of Transformer architecture [12]. These model advancements have proved incredibly capable of generating human-like text, but where they are not so effective is in navigating domain-specific questions that call upon the need for specialized knowledge.

RAG methodology fuses information retrieval and generative models, and has shown enormous potential in enhancing the performance of LLMs for Q&A tasks. RAG systems create responses that are contextually rich and informative. These systems have been adopted in the medical field, and are still in burgeoning stages.

MedGraphRAG is a graph-based Retrieval-Augmented Generation (RAG) framework designed to improve the reliability of LLMs in the medical domain by generating evidence-based results. It employs a hybrid static-semantic document chunking approach to improve context capture, and constructs a three-tier hierarchical graph linking entities to foundational medical knowledge. These meta-graphs are merged semantically to create a global graph, enabling precise retrieval and response generation using the U-retrieve method. Comprehensive evaluations demonstrate that MedGraphRAG outperforms state-of-the-art models on medical Q&A benchmarks, providing source documentation to improve the reliability of medical LLMs. The MEDRAG system was developed to evaluate how different retrievers and corpora impact performance in medical Q&A tasks [13]. Similarly, the Self-BioRAG project [14] adapted Self-RAG technology [15] for medical applications, enhancing it with reflective tokens to improve retrieval timing, assess the relevance and supportiveness of retrieved documents, and evaluate the quality of generated answers. The Bailicai framework [16], a RAG that has been meticulously refined to address challenges in medical applications reduces data noise by incorporating domain-specific medical knowledge, employs self-knowledge boundary identification to optimize RAG innovation. This is enabled by the construction of task-specific datasets so models can be optimized for specific tasks. They also employ significant iterative prompt engineering. For this research, these advancements significantly reduced hallucinations and improved the performance of LLMs in the medical domain.

Data

In this section we break down the details of the HRA biomarker dataset. It contains a collection of biomarkers organized hierarchically. There are 2,030 rows structured under a root node labeled "biomarkers," which indicates the overarching category of the data. It is a flat hierarchical structure. While the dataset is organized under a hierarchical framework, most biomarkers are directly categorized under their respective parent nodes, resulting in a flat structure at this level. Child relationships are sparsely defined, with only six entries having non-empty child attributes, limiting their usability for downstream tasks. Biomarkers are well-categorized into distinct parent categories, with "Gene" and "Protein" being the most dominant groups. The dataset includes a diverse set of biomarker labels and synonym labels, providing rich contextual information for each biomarker. Every entry in the dataset is a "node", and represents an individual biomarker.

Schema

- **@id and id:**
    - Unique identifiers for each biomarker are represented as URLs.
- **@type:**
    - Specifies the node type, which is uniformly labeled as "OntologyTreeNode" -- these entries belong to an ontology or classification system.
- **parent:**
    - This indicates the flavor of the biomarker. There are six unique parent categories with the following distribution:
        - **Gene:** 1,590 entries
        - **Protein:** 418 entries
        - **Lipids:** 9 entries
        - **Proteoforms:** 5 entries
        - **Biomarkers:** 5 entries
        - **Metabolites:** 2 entries
    - **Number of null values:** 0.
- **children:**
    - A list of child nodes associated with the biomarker. Most rows have an empty list, with only six rows containing non-empty values, suggesting minimal utility of this attribute.
- **synonymLabels:**
    - Represents synonyms for each biomarker. There are 462 unique values and 362 non-empty rows.
- **label:**
    - A human-readable identifier for the biomarker, often corresponding to a gene symbol (e.g., "A2M," "ABCA1"). There are 1,930 unique values.

Methods

We used a pre-trained Eleuther AI LLM model combined with RAG embeddings to answer questions about the HRA documentation. The Eleuther GPT-Neo 1.3B is a transformer model using a replication of the GPT-3 architecture. The model was trained on a large scale dataset by Eleuther AI as a masked autoregressive language mode and used cross-entropy loss [17].

Our Eleuther Model notebook first loads and parses the biomarker data to extract all relevant information from the JSON file. We then parse the information and combine it into a dataframe to encode the embeddings. The model is initiated by using a sentence transformer. A FIASS index is created to implement the retrieval process. We then set up the LLM tokenizer as the

Eleuther GPT-Neo model. Our model is initiated by asking a question into a function which will insert the question into the prompt along with context and run the model to generate an answer.

We created two different prompts to test the model on. The first prompt was more generic with less structure on how to answer:

```
f"The following is some context from a document:\n{context}. The document is a hierarchical structure of biomarkers.\n\n"
      f"You are a medical expert specializing in biomarker research. Using the context above, answer the following question:\n{question}\n\n"
      f"Answer (use the context as part of the answer):"
```

The second prompt had more detail and gave a structure on how to answer the question:

```
f"The following is some context from a document:\n{context}. The document is a hierarchical structure of biomarkers.\n\n"
      f"You are a medical expert specializing in biomarker research.\n\n"
      f"An example question is, 'What is ABCA1?'\n\n"
      f"Example answer: 'ABCA1 is a gene that is part of the biomarker structure with no children, synonymLabels and can be found at http://identifiers.org/hgnc/29.'\n\n"
      f"Using the context above, answer the following question:\n{question}\n\n"
      f"Answer (use the context as part of the answer):\n\n"
```

Our hypothesis was that the more detailed prompt would serve the RAG process better by giving it a direction in how to answer and give more specific answers while the first may give more in-depth answers.

We decided to evaluate each prompt and model by asking both the same question and rating the response of the model using a scale from 1-3 of relevancy. This will be explained further in the following section.

Results

To evaluate performance of our two models, we applied a method of human evaluation by scoring metrics from 1-3 for relevancy. We evaluated a total of 45 prompts for each model.

1. Irrelevant
2. Somewhat relevant
3. Highly relevant

Although a smaller scoring range can limit granularity of the analysis, we finalized this methodology to avoid subjective variability. Many of the calculations below were performed on a python notebook that can be found on our GitHub page here: https://github.com/paytonncourt96/Deep_Learning/blob/main/Code%20Samples/Statistical_Analysis.ipynb with n=45 for sample size.

Mean and Standard Deviation:

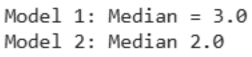
Sample Mean:

$$\bar{x} = \frac{1}{n}\sum_{i=1}^{n} x_i$$

$$\bar{x}_1 = 2.42$$

$$\bar{x}_2 = 1.73$$

Median:



```
Model 1: Median = 3.0
Model 2: Median 2.0
```

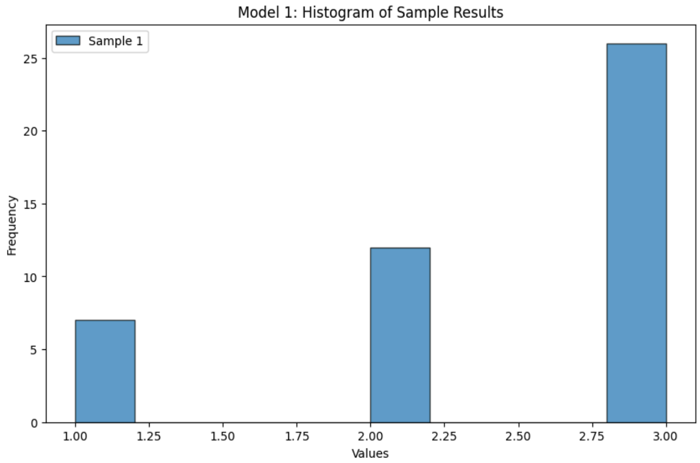
Sample Standard Deviation

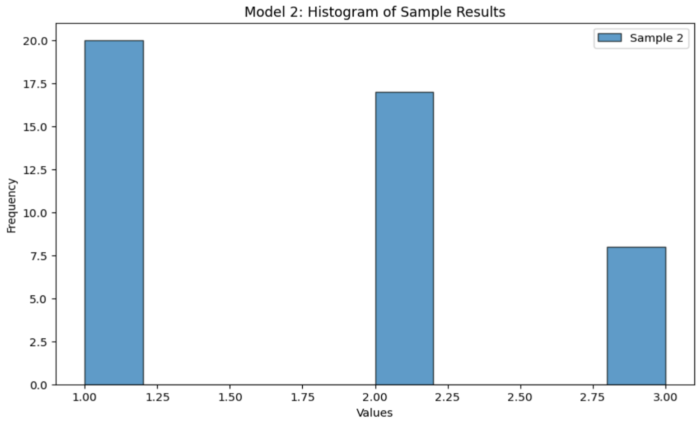
$$s = \sqrt{\frac{1}{n-1}\sum_{x=1}^{n} (x_i - \bar{x})^2}$$

$$s_1 = 0.75$$

$$s_2 = 0.75$$

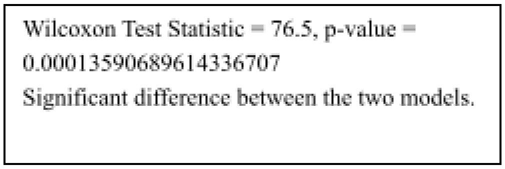
Histogram of Results:





Wilcoxon-Signed Rank Test [18]:
The statistical model chosen to evaluate model performance is the Wilcoxon-Signed Rank test. Since our model approaches are related by both the model itself and the same human reviewers, this model was felt as the best approach to decide if the models differed enough for any future work. The test was performed in python:

Result:



```
Wilcoxon Test Statistic = 76.5, p-value =
0.00013590689614336707
Significant difference between the two models.
```

From this result, we can reasonably conclude there was a large enough difference in performance between our two models to choose the highest performing prompt.

Note:
Although model 2 performed lower within the statistical evaluations, it is worth noting that this model provided answers related more to the actual biomarker dataset itself. This was an overall goal, to provide context from the dataset and supplement other resources.

Discussion/Future Work

If time and resources were to be permitted, a future goal on this project would be to incorporate a hybrid large language model. The approach would utilize an open-source model with conjunction of custom training and evaluation. This methodology would allow for more fine tuning and prompt engineering to achieve optimal results. An attempt on the hybrid model was made and can be found in the github link here: https://github.com/paytonncourt96/Deep_Learning/blob/main/Code%20Samples/finetunable_eleuthermodel.py. This code provides a hybrid approach on GPT-Neo from Eleuther[17] with further emphasis on fine-tuning and training. However, due to resource limitations within Google Colab and model evaluation efficiency, we decided to table this approach.

Training:

In the attached code, a draft train set was created. Ideally, this would be made by or under the advice of subject matter experts. A starting point of around 100 training questions and answers examples would be a sufficient first pass. The train set would help guide the model to find more meaningful insights from the dataset contexts. This would also aid the model in understanding the type of results desired.

Fine-tuning:

Although GPT-neo is trained on large, diverse sets, the parameters are not established to tackle our specific task. In order to modify GPT-neo's customizations, training arguments with defined parameters outside of the pre-defined ones set by neo would be defined. The first piece of which would be a low-learning rate. This would aid the model in not passing on too quickly from any pre-trained knowledge and avoid overfitting as changes occur. Another important fine-tuning piece is batch size. This helps limit computing resources as the model is trained and is helpful to customize since our biomarker dataset is somewhat small.

Prompt Engineering:

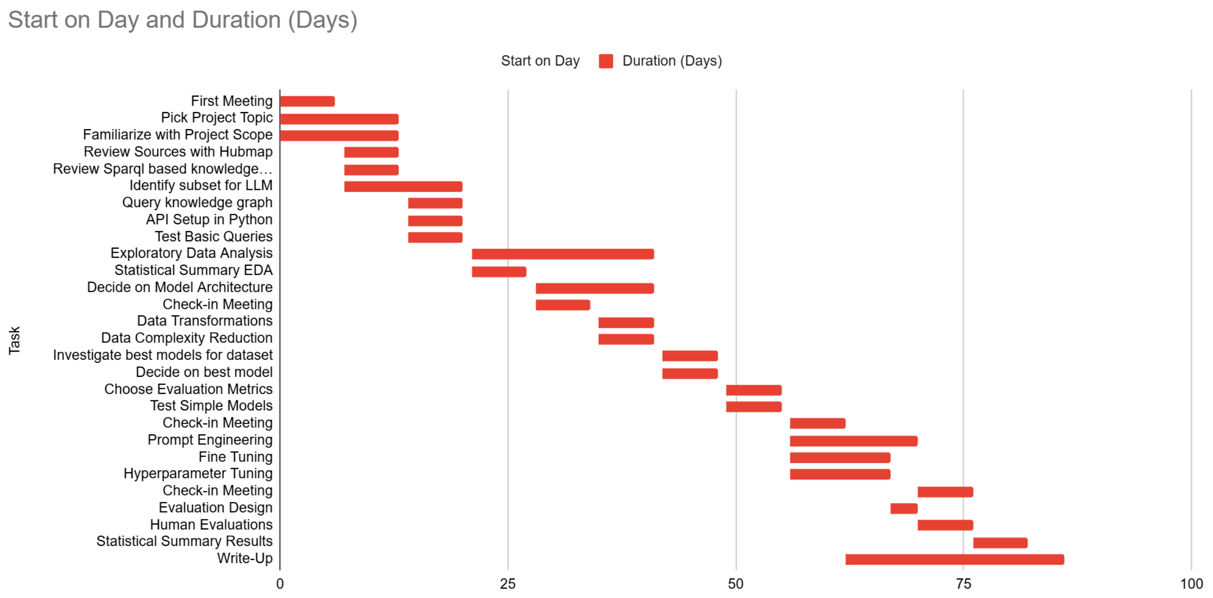
In the current phase, we evaluated the results by comparing the model's performance on two defined prompts. This approach provided valuable insights into:
Strengths: Identifying which aspects of the prompts generated accurate, highly relevant, or specific responses.
Weaknesses: Pinpointing elements of the prompts that led to ambiguity, misinterpretation, or irrelevant responses.
While this comparison offers a clear understanding of how the prompts influence model behavior, the next step lies in utilizing these insights for an improved prompt design.

Conclusion:

By employing a methodical approach to hybrid large language model development from leveraging an open-source framework while integrating personalized training techniques, custom fine-tuning parameters, and insights from prompt engineering, a more robust and adaptable model can be achieved. Since the HUBMAP hosts far more than just biomarker data, we could apply this model improvement to that of the other contents hosted within the knowledge graph database.

Gantt Chart

The Gantt chart outlines the schedule of work applied to our project, highlighting key phases and milestones, while our team maintained constant communication and alignment through Slack.



Literature

1. Siyun Zhao, Yuqing Yang, Zilong Wang, Zhiyuan He, Luna K. Qiu, Lili Qiu. Retrieval Augmented Generation (RAG) and Beyond: A Comprehensive Survey on How to Make your LLMs use External Data More Wisely. https://arxiv.org/abs/2409.14924
2. Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, Haofen Wang. Retrieval-Augmented Generation for Large Language Models: A Survey. arXiv:2312.10997 [cs.CL]
3. Zooey Nguyen, Anthony Annunziata, Vinh Luong, Sang Dinh, Quynh Le, Anh Hai Ha, Chanh Le, Hong An Phan, Shruti Raghavan, Christopher Nguyen. Enhancing Q&A with Domain-Specific Fine-Tuning and Iterative Reasoning: A Comparative Study. arXiv:2404.11792v2 [cs.AI] 19 Apr 2024

4. Quang Nguyen, Duy-Anh Nguyen, Khang Dang, Siyin Liu, Khai Nguyen, Sophia Y. Wang, William Woof, Peter Thomas, Praveen J. Patel, Konstantinos Balaskas, Johan H. Thygesen, Honghan Wu, Nikolas Pontikos. Advancing Question-Answering in Ophthalmology with Retrieval-Augmented Generation (RAG): Benchmarking Open-source and Proprietary Large Language Models. https://doi.org/10.1101/2024.11.18.24317510
5. Junde Wu. Medical Graph RAG: Towards Safe Medical Large Language Model via Graph Retrieval-Augmented Generation. https://arxiv.org/html/2408.04187v1
6. Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, Douwe Kiela. Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. arXiv:2005.11401v4 [cs.CL] 12 Apr 2021
7. Kunal Sawarkar, Abhilasha Mangal, Shivam Raj Solanki. Blended RAG: Improving RAG (Retriever-Augmented Generation) Accuracy with Semantic Search and Hybrid Query-Based Retrievers. arXiv:2404.07220
8. Börner, K., Teichmann, S.A., Quardokus, E.M. *et al.* Anatomical structures, cell types and biomarkers of the Human Reference Atlas. *Nat Cell Biol* 23, 1117–1128 (2021). https://doi.org/10.1038/s41556-021-00788-6
9. Tom B. Brown et al. Language Models are Few-Shot Learners. 2020. arXiv: 2005.14165 [cs.CL].
10. Jacob Devlin et al. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. 2018. arXiv: 1810.04805 [cs.CL].
11. Yinhan Liu et al. RoBERTa: A Robustly Optimized BERT Pretraining Approach. 2019. arXiv: 1907. 11692 [cs.CL].
12. Ashish Vaswani et al. Attention Is All You Need. 2017. arXiv: 1706.03762 [cs.CL].
13. Guangzhi Xiong, Qiao Jin, Zhiyong Lu, and Aidong Zhang. Benchmarking retrieval-augmented generation for medicine. arXiv preprint arXiv:2402.13178, 2024.
14. Minbyul Jeong, Jiwoong Sohn, Mujeen Sung, and Jaewoo Kang. Improving medical reasoning through retrieval and self-reflection with retrieval-augmented large language models. Bioinformatics, 40(Supplement 1):i119–i129, 2024.

15. Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. Self-rag: Learning to retrieve, generate, and critique through self-reflection. arXiv preprint arXiv:2310.11511, 2023.
16. Lui Yongbin, Long Cui, Ouyang, Chunping, Yu Ying. Bailicai: A Domain-Optimized Retrieval-Augmented Generation Framework for Medical Applications arXiv:2407.21055v1 [cs.CL] 24 Jul 2024
17. Hugging Face. EleutherAI/gpt-neo-1.3B. https://huggingface.co/EleutherAI/gpt-neo-1.3B. 3 May 2023
18. Doshi R, Amin KS, Khosla P, Bajaj SS, Chheang S, Forman HP. Quantitative Evaluation of Large Language Models to Streamline Radiology Report Impressions: A Multimodal Retrospective Analysis. Radiology. 2024 Mar;310(3):e231593. doi: 10.1148/radiol.231593. PMID: 38530171.