MediMind - Automated Radiology Report Generation using RAG

Nitika Jain
Khoury College of Computer Science
Northeastern University
Boston,USA
jain.nit@northeastern.edu

Shikha Tiwari
Khoury College of Computer Science
Northeastern University
Boston, USA
tiwari.shi@northeastern.edu

***Abstract*—Medical imaging is frequently used in clinical practice for diagnosis and treatment. Writing reports can be a challenging task for trainee radiologist who may not have much experience in the field. Also, it can be tiresome as radiologist need to write multiple reports in a day which could lead to errors in report. With, the use of artifcial intelligence and specifically LLM models trained on medical data we can utilize those models to generate radiology reports. But LLM models are also prone to hallucinations and can be generic, to reduce this drawback we introduce MediMind RAG based radiology report generation system where we utilized PubMedCLIP multimodal embeddings model to generate image vector embeddings. User can provide a query image; system will run the similarity search and fetch top k impressions which will be used as context and are passed to LLM model along with prompt to provide radiology summary. We evaluated our system using LLM as a Judge wherein we measured the relevance, accuracy and conciseness of the generated response with respect to ground truth and achieved a score of 3, on a scale of 5.**

Keywords—LLM (Large Language Model), RAG (Retrieval Augmented Generation), PubMedCLIP, BioMistral

I. INTRODUCTION

Medical imaging is frequently used in clinical practice for diagnosis and treatment. With the exponential growth of radiological imaging data and less availability of trained radiologists, it increases workload of radiologist and can become tiresome for them to make correct diagnosis and put the findings down accurately in the report. As an effect, fatigue may creep in and it could have serious consequences as we are dealing with critical health data and incorrect diagnosis may lead to wrong treatment being administered to the patient that may lead to serious health consequences in some cases. Our contribution to resolve this challenge, called MediMind, is an automated radiology report generation system which aims to produce a coherent and accurate summary of Chest-X-rays similar to that of an experienced radiologist could produce. We tried to solve a Retrieval problem where we utilized IU-XRAY dataset collected by Indian Univeristy Medical Department where we will take advantage of set of chest x-ray image and diagnostic details associated with those images. This approach can utilize verified data and report generated by professional radiologist and generate radiology impression of the provided query image. To generate the summary impression, we developed a RAG system utilizing pretrained LLM models like PubMedCLIP and BioMistral quantized version. Since these models are already trained on pubmed medical data it fits well with our use case. For RAG System, we had three main components. Indexing, where we used IUXRAY data and generated image vector embeddings of train X Ray images using PubMedCLIP encoder, since PubMedCLIP model is trained on ROCO dataset it will generate accurate X Ray image embeddings. Once we got the embeddings we stored them in Qdrant Cloud Vector database for easy retrieval. Then, when user upload an image we will utilize same encoder model to generate image embeddings and generate then will do a cosine similarity search and export top k impression from Qdrant database. Next, we will pass on this top impression as a context along with our prompt to a LLM Model to genrate radiology summary. We were motivated by the Retrieval Augmented Generation(RAG) work by Lewis et al.(2020) [1] and Chest X Ray report generation work done by Endo Mark et al.(2021) [2].

II. BACKGROUND

A. Retrieval Augmented Generation

RAG system retrieves more information and data from external knowledge bases through information retrieval, combining the capabilities of traditional large language models (LLMs) with information retrieval .It combines the strengths of traditional information system with the capabilities of generative Large Language models. LLM are limited to their pre trained data which can eventually lead to outdated and inaccurate responses, LLMs are also prone to hallucinations and provide factually incorrect information. As we were dealing with medical image data more specifically radiology summaries, where we can not risk inaccurate information and hallucinations.

We were highly motivated by Xia, Peng, et al.(2024) [3] where they introduced a verstatile multimodal RAG System involving domain- aware retrieval mechanism designed to handle different domains of medical images. We took inspiration from this and decided to use encoder which is specifically trained on medical data to generate embeddings for our dataset for efficient retrieval.

B. Vector Database

A vector database is a collection of data stored as mathematical representations. Vector databases make it easier for machine learning models to remember previous inputs, allowing machine learning to be used to power search, recommendations, and text generation use-cases. Data can be identified based on similarity metrics instead of exact matches, making it possible for a computer model to understand data contextually.

For scope of the project we used Qdrant Vector Database. Qdrant is an open-source and fully managed high-performance, massive-scale Vector Database. The vector search engine provides a production-ready service with a convenient API to store, search, and manage vectors with an additional payload. Qdrant is tailored to extended filtering support on additional metadata fields, which can be stored as payload along with vector embeddings. With Qdrant,

embeddings and neural network encoders can be turned into full-fledged applications for matching, searching, recommending, and many more solutions to make the most of unstructured data. We stored the image vector embeddings along with the respective radiological summary so when we perform similarity search with the query image, we can retrieve the summary also.

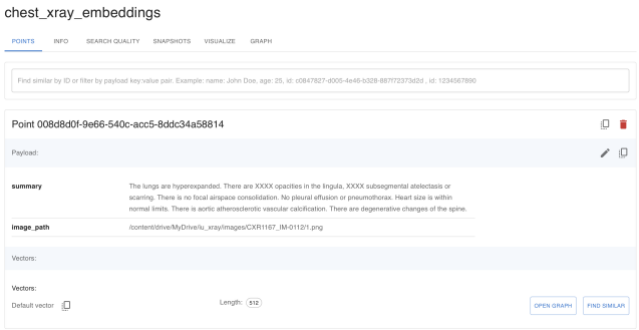

Fig. 1. QDRANT Vector Database snapshot

C. *PubMedCLIP Model*

We have utilized PubMedCLIP model encoder to generate image vector embeddings. PubMedCLIP was trained on the Radiology Objects in COntext (ROCO) dataset, a large-scale multimodal medical imaging dataset. The ROCO dataset includes diverse imaging modalities (such as X-Ray, MRI, ultrasound, fluoroscopy, etc.) from various human body regions (such as head, spine, chest, abdomen, etc.) captured from open-access PubMed articles.

PubMedCLIP was trained for 50 epochs with a batch size of 64 using the Adam optimizer with a learning rate of 10−5.

D. *BioMistral Model*

We have utilized 4 bit quantized version of BioMistral Model as our LLM model to generate radiological summary. BioMistral is an open-source LLM tailored for the biomedical domain, utilizing Mistral as its foundation model and further pre-trained on PubMed Central. Model was evaluated on a benchmark comprising 10 established medical question-answering (QA) tasks in English. Also, to address the limited availability of data beyond English and to assess the multilingual generalization of medical LLMs, it automatically translated and evaluated this benchmark into 7 other languages.

III. Related Work

A. *Ranjit, Mercy, et al. Retrieval augmented chest x-ray report generation using openai gpt models[4]*

Our Project was highly inspired by Ranjit Mercy, et al. where they have proposes Retrieval Augmented Generation (RAG) as an approach for automated radiology report writing that leverages multimodally aligned embeddings from a contrastively pretrained vision language model for retrieval of relevant candidate radiology text for an input radiology image and a general domain generative model like OpenAI text-davinci-003, gpt-3.5-turbo and gpt-4 for report generation using the relevant radiology text retrieved.

They utilized MIMIC CXR Dataset, for embeddings model and used ALBEF model from CXR-ReDonE to generate vision language aligned embeddings for a database of radiology reports. As the image and text embeddings were aligned during the contrastive pre-training, the most relevant text radiology text (reports or sentences) is retrieved for an input x-ray image based on the similarity of the embeddings. A consolidated radiology report impression is generated from the filtered set of records using the OpenAI text-davinci-003, gpt-3.5-turbo and gpt-4 models.

We experimented with different models and did not use the same encoder and LLM models to generate repsonse. Moreover, in this they matched text embeddings of source and image embeddings of input image. We modified and experimented with image embeddings of source and input image.

B. *MRScore: Evaluating Radiology Report Generation with LLM-based Reward System [5]*

In this paper they introduced MRScore, an automatic evaluation metric tailored for radiology report generation by leveraging Large Language Models (LLMs). Conventional NLG (natural language generation) metrics like BLEU are inadequate for accurately assessing the generated radiology reports, as systematically demonstrated by our observations within this paper. To address this challenge, they collaborated with radiologists to develop a framework that guides LLMs for radiology report evaluation, ensuring alignment with human analysis. Their framework includes two key components: i) utilizing GPT to generate large amounts of training data, i.e., reports with different qualities, and ii) pairing GPT-generated reports as accepted and rejected samples and training LLMs to produce MRScore as the model reward.

We were inspired by this paper and for evaluation we have utilized LLM as a judge to classify the impression generated based on completeness, relevance, and accuracy. We specifically used Sonnet 3.5. With each iteration we analyzed the output and modified the prompt.

C. *Additional Methods and Approaches*

Alternative methods and approach which we could have done was utilizing multiple LLM models and comparing results across different settings. There are multiple paid models which we believe could have performed better but we sticked to utilize the open-source models. Moreover, for the LLM model we used is quantized version of BioMistral as we had limited resource.

For evaluation, we would also do a human evaluation who is expert in the field and generate summaries daily. They can better judge the system as LLM models also are comparing with the metadata and baseline summaries where there can be a scenario where some information could have been missed by LLM model which will be highlighted by professional radiologist or a pulmonologist.

We wanted to include Contextual Augmentation where we would include additional patient metadata like age, gender, clinical history or related image studies to provide richer context to the model. We were not able to do this as we just had radiological summary available.

IV. Project Description

A. Dataset Description

We have used IU-XRAY Images for our project. Open-i has a collection of chest X-Ray Images from the Indiana University hospital network. Data contains two folders, one for X-ray Images and the other for the report of radiography.

The dataset had a json summary which had Image id, radiology report and image path. For image there were mainly two types of images front and side.

Json Summary: {"id": "CXR899_IM-2407", "report": "The lungs are clear. There is no focal consolidation, pleural effusion, or pneumothorax. The Heart and mediastinum are normal size and shape. XXXX and soft tissues are unremarkable.", "image_path": ["CXR899_IM-2407/0.png", "CXR899_IM-2407/1.png"], "split": "train"}

Fig. 2. CXR 899_IM-2407/0.png Front Chest X-ray

Fig. 3. CXR899_IM-2407/1.png Side Chest X-ray

B. System Architecture

Our Project has 2 main components:

1) Indexing

The indexing process in our system starts by organizing a dataset of chest X-ray images paired with their corresponding summary impressions, which are concise radiological findings. Each image is then passed through a pretrained model, PubMed-CLIP, to generate vector embeddings. These embeddings are essentially numerical representations of the images, capturing important medical details in a way that a computer can understand.

Once the embeddings are created, they're stored in a vector database Qdrant, which is designed to handle large amounts of high-dimensional data. It also makes it easy to perform fast similarity searches. During this phase, we link each image's embedding to its corresponding summary impression. This creates a well-structured index that allows the system to quickly find the most relevant impressions when a new X-ray image is provided. This indexing step is crucial because it ensures the system can retrieve accurate and meaningful context during report generation.

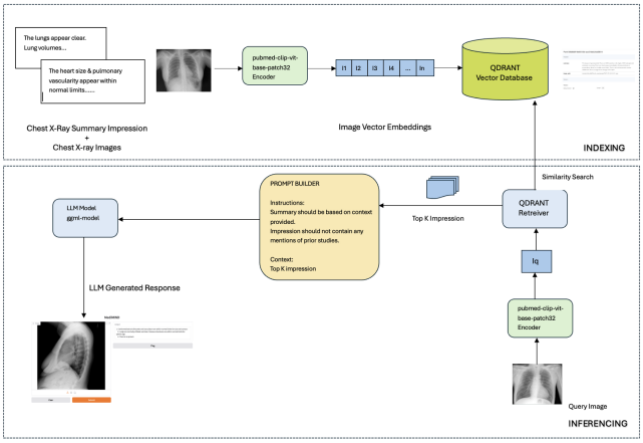

Fig. 4. System Design

2) Inferencing

The inferencing phase begins with the user providing a query X-ray image. This image is processed through the pubmed-clip-vit-base-patch32 encoder, which generates a high-dimensional vector embedding representing the visual and contextual features of the image. These embeddings are then passed to the QDRANT vector retriever, which performs a similarity search against the indexed embeddings created during the indexing phase. The retriever identifies the top K most similar impressions from the database.

These top K impressions are sent to the prompt builder, which crafts a structured input for the LLM model. The prompt builder includes specific instructions, such as ensuring that the generated summary is based solely on the provided context and avoids any references to prior studies. The crafted prompt, along with the contextual impressions, is then passed to the LLM model, which generates a detailed radiology summary of the input image. The system outputs the generated report with high accuracy, capturing relevant medical details of the X-ray and presenting them in an easily interpretable format for clinicians.

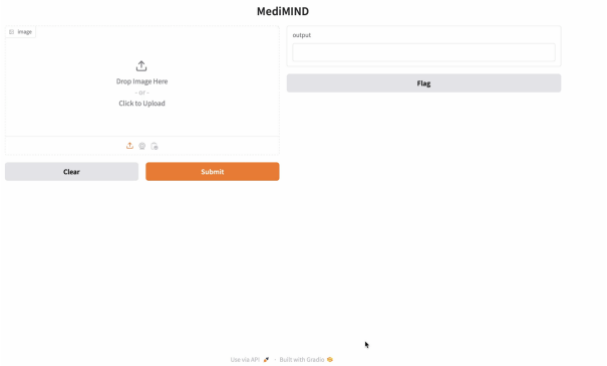

Fig. 5. System Stage 1 – where user will upload the image and submit to generate radiology summary

Fig. 6. System Stage 2 – System generated summary

V. Emperical Results

A. *Retrieval Evaluation*

To assess retrieval accuracy, we tested the vector database by embedding 50 test images and performing a semantic search using cosine similarity. The objective was to retrieve the correct reports associated with these images based on their embeddings. Using the PubMed/BERT model, the semantic search successfully matched all 50 image embeddings with their corresponding reports, achieving 100% retrieval accuracy. This evaluation ensured that the relevant textual information corresponding to the user's query image embeddings was accurately fetched from our image-text paired data.

B. *Generator Evaluation*

The evaluation of the generator focused on assessing the completeness, coherence, relevance, and accuracy of the generated responses. Since we are working with a medical dataset, evaluating the smallest nuances in the generated outputs requires domain expertise. In the absence of a domain expert, we employed a large language model (LLM), specifically Claude Sonnet 3.5, as a judge.

The LLM was tasked with measuring the quality of the generated response by comparing it to the original report in our dataset (base report/metadata). The evaluation process assessed whether the generated outputs aligned with the expected standards of medical reporting, ensuring that the responses were relevant, logically structured, and comprehensive. By incorporating these evaluation methods, we were able to rigorously test the retrieval and generation components of our system, ensuring that it delivers high-quality, accurate, and contextually appropriate responses.

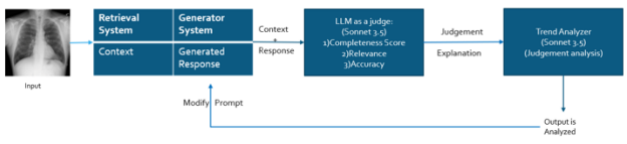

Fig. 7. Evaluation Architecture

For a dataset of 50 image-report pairs, we provided the original report (from our JSON file) and the generated response from our RAG system to an LLM (Claude 3.5 Sonnet). Using a custom-designed prompt with detailed judgment criteria, the LLM scored each response as 0 or 1, accompanied by an explanation for the assigned score.

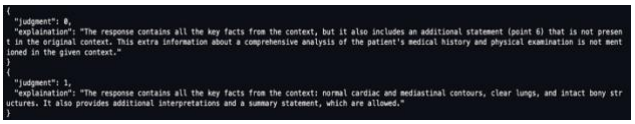

Fig. 8. Judgment output

After gathering explanations for all 50 samples, we passed them to another instance of Claude 3.5 Sonnet with a prompt designed to identify common patterns and issues in the scores.

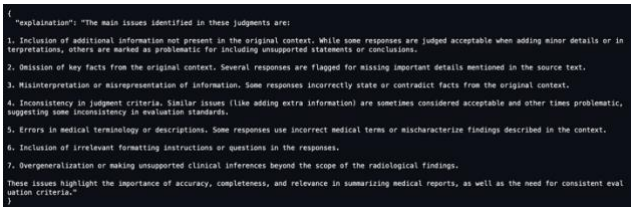

Fig. 9. LLM as a Judge output evaluation

The evaluation process also included iterative refinement of the RAG system's prompt based on feedback, with the goal of improving the accuracy of generated responses. The concise set of rules derived from the evaluation was instrumental in refining the system and enhancing its performance. In the first testing iteration, the RAG system achieved an accuracy of 72%. Accuracy was calculated as the proportion of correct responses (scored as 1) assigned by the LLM acting as a judge.

The formula used was:$Accuracy_1$ = Total Number of 1's / Total Number of Samples $Accuracy_1 = 36/50 = 72\%$.This score highlighted areas for improvement, which were addressed by refining the prompt based on rules and insights provided by the LLM's feedback. After refining the RAG system's prompt based on rules extracted from the LLM's feedback, we conducted a second iteration. This time, the LLM gave a score of 1 to 40 out of 50 samples, yielding an improved accuracy of 80%, reflecting an 8% improvement.

These results demonstrate that our RAG system now achieves a confidence level of 80% in generating accurate responses.

C. *Evaluation on a sacle 1-5*

To further evaluate the quality of the generated responses, we prompted the LLM, acting as a judge, to provide an overall score between 1 and 5 for each test sample based on three key criteria: relevance, accuracy, and conciseness. Relevance assessed whether the response directly addressed the query, accuracy evaluated the correctness of the information provided, and conciseness examined whether the response was free from unnecessary details.

Each sample in the test dataset was scored accordingly, and we compared the average scores of two systems: the LLM-generated responses without the RAG system and the RAG-generated responses.

The responses generated by the standalone LLM, without utilizing the RAG system, achieved an average score of 1.0 across all 50 test samples. In contrast, the

RAG-generated responses attained an average score of 3.0, indicating a significant improvement. This comparison clearly demonstrates the effectiveness of the RAG system in producing responses that are more relevant, accurate, and concise. By integrating the retrieval capabilities of the RAG system, we ensured that the responses were contextually informed and better aligned with the query. These results highlight the superior performance of the RAG system over standalone LLM-generated responses in addressing the evaluation criteria.

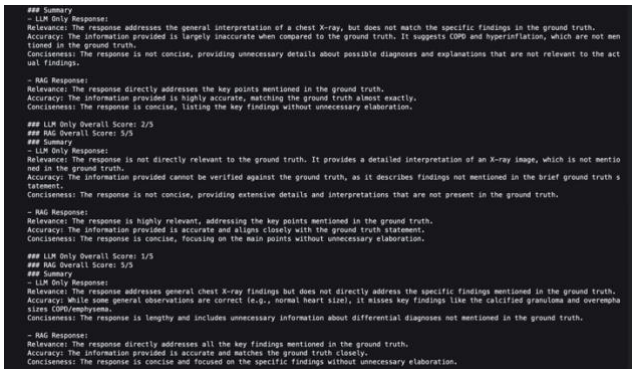

Fig. 10. LLM as a Judge output evaluation on a scale of 1-5

VI. Broader Implication

The broader implications of this project, which combines Retrieval-Augmented Generation (RAG) with medical image analysis to improve chest X-ray report generation, are far-reaching, especially in the healthcare sector. One of the most significant impacts is the potential to enhance diagnostic accuracy and assist healthcare professionals in making better, faster decisions. By automating the generation of reports from X-ray images, the system can reduce human error and provide timely information to medical practitioners, which is crucial in high-pressure situations.

Beyond the immediate healthcare benefits, this project could have a major impact on accessibility to healthcare services. In regions with limited access to specialized radiologists, the system can help bridge the gap by providing reliable diagnostic support without the need for an expert to review each image. This can potentially lead to faster diagnoses, especially for conditions like pneumonia, where timely treatment is critical. In this way, the project could help alleviate some of the pressure on healthcare workers and make healthcare more accessible to underserved populations.

However, there are also important societal and ethical considerations. One concern is the potential for over-reliance on AI systems in healthcare. While the system is designed to assist rather than replace healthcare professionals, there's a risk that mistakes made by the AI could go unnoticed, leading to incorrect diagnoses. This highlights the need for continuous monitoring, regular validation, and strong regulatory oversight to ensure the system's reliability. Additionally, handling sensitive medical data raises privacy concerns, making it essential to implement robust data protection measures to safeguard patient confidentiality.

VII. Conclusion

We achieved an average score of 3 on a scale of 5 and an accuracy of 80% in our RAG system by leveraging LLMs as both a judge and an error analyzer. Through this project, we gained a comprehensive understanding of the end-to-end pipeline, from data collection and embedding creation to implementing and evaluating a RAG system. A key takeaway was that there is no "one-size-fits-all" solution with LLMs; success requires experimentation with different models, variants, and processes to identify the best fit. Given the rapidly evolving landscape of LLMs and techniques like RAG, staying updated with current research is essential for informed decision-making.

With more time, we would explore additional LLM models within our RAG system and compare their performance as judged by the LLM evaluator. We would also experiment with fine-tuning and integrating those models into our pipeline. While this project focused on image embeddings, we plan to incorporate image and text embeddings in the future to enhance retrieval and response generation. For the future students of this course, we recommend exploring mutiple sets of data in different domains which would help ustilizing LLM to help solve multiple problems across domains and making an impact. LLMs are a dynamic research domain; limiting yourself to 2-3 models won't suffice. Experiment widely, remain adaptable, and embrace the iterative nature of research to uncover the best solutions.

VIII. Github

Please find our project at following github repo:
https://github.com/ShikhaTiwari1809/MediMIND

References

[1] Lewis, Patrick, et al. "Retrieval-augmented generation for knowledge-intensive nlp tasks." Advances in Neural Information Processing Systems 33 (2020): 9459-9474

[2] Endo, Mark et al. "Retrieval-Based Chest X-Ray Report Generation Using a Pre-trained Contrastive Language-Image Model." ML4H@NeurIPS (2021).

[3] Xia, Peng, et al. "Mmed-rag: Versatile multimodal rag system for medical vision language models." arXiv preprint arXiv:2410.13085 (2024)

[4] Ranjit, Mercy, et al. "Retrieval augmented chest x-ray report generation using openai gpt models." Machine Learning for Healthcare Conference. PMLR, 2023.

[5] Liu, Yunyi, et al. "MRScore: Evaluating Radiology Report Generation with LLM-based Reward System." arXiv preprint arXiv:2404.17778 (2024)