Degree Project in Computer Science and Engineering

Second cycle, 30 credits

Evaluation of Retrieval-Augmented Generation in Medical Question Answering Tasks

TANGYUJUN HAN

Evaluation of Retrieval-Augmented Generation in Medical Question Answering Tasks

TANGYUJUN HAN

Master's Programme, Communication Systems, 120 credits
Date: December 9, 2024

Supervisor: Amirhossein Layegh Kheirabadi
Examiner: Amir H. Payberah
School of Electrical Engineering and Computer Science
Swedish title: Utvärdering av hämtningsförstärkt generation i medicinska fråge-svar uppgifter

Abstract

Recent developments and changes in Large Language Models (LLMs) have great potential for application in the field of medical question answering (QA), particularly through Retrieval-Augmented Generation (RAG) systems. These systems address challenges in providing reliable and personalized medical information by integrating authoritative sources. However, evaluating their performance remains a critical challenge, especially in sensitive medical contexts where accuracy is critical. Current evaluation techniques often rely on heavy human annotations, making the process time-consuming and labor-intensive. While using LLMs as evaluators has been proposed as an alternative to reduce the manual workload, its reliability remains questionable.

This thesis introduces a new evaluation method to solve this problem, tested by constructing various RAG systems, including Naive RAG and Hypothetical Document Embeddings (HyDE) RAG. The evaluation leverages two different LLMs and is based on a benchmark dataset specifically designed for yes/no medical questions, with an LLM-only system serving as the baseline. Metrics used for evaluation include Accuracy, Precision, Recall, F1 score, Mean Accuracy (MAP), and Mean Reciprocity Rating (MRR) to measure retrieval and generation performance comprehensively. In addition, the study explored the impact of different search relevance thresholds and different models on the RAG system, providing insights for further optimization.

The experimental results show that RAG systems greatly improve the accuracy and reliability of medical information retrieval compared to baseline models. The choice of retrieval relevance thresholds and the selection of different LLMs also impact the performance of RAG systems. The paper proposes a robust evaluation method for RAG systems in medical QA and lays the foundation for extending this method into other knowledge-intensive domains. Such reliable evaluations will contribute to developing more effective and reliable medical QA systems, benefiting both healthcare providers and patients.

Keywords

Medical Question Answering, Large Language Models, Retrieval-Augmented Generation, Evaluation

Sammanfattning

Den senaste tidens utveckling och förändringar inom Stora språkmodeller (LLMs) har stor potential för tillämpning inom området medicinsk frågesvar (QA), särskilt genom Retrieval-Augmented Generation (RAG) system. Dessa system hanterar utmaningar när det gäller att tillhandahålla tillförlitlig och personlig medicinsk information genom att integrera auktoritativa källor. Att utvärdera deras prestanda är dock fortfarande en stor utmaning, särskilt i känsliga medicinska sammanhang där noggrannhet är avgörande. Nuvarande utvärderingstekniker förlitar sig ofta på tunga mänskliga kommentarer, vilket gör processen tidskrävande och arbetsintensiv. Att använda LLM:er som utvärderare har föreslagits som ett alternativ för att minska den manuella arbetsbelastningen, men dess tillförlitlighet är fortfarande tveksam.

Denna avhandling introducerar en ny utvärderingsmetod för att lösa detta problem, testad genom att konstruera olika RAG-system, inklusive Naive RAG och Hypothetical Document Embeddings (HyDE) RAG. Utvärderingen baseras på en referensdatauppsättning som är särskilt utformad för medicinska ja/nej-frågor, med ett LLM-only-system som fungerar som baslinje. Mätvärden som används för utvärdering inkluderar noggrannhet, precision, återkallande, F1 poäng, genomsnittlig noggrannhet (MAP) och genomsnittlig ömsesidighet (MRR) för att på ett heltäckande sätt mäta prestanda för hämtning och generering. Dessutom undersökte studien effekterna av olika tröskelvärden för sökrelevans och olika modeller på RAG-systemet, vilket gav insikter för ytterligare optimering.

De experimentella resultaten visar att RAG-systemen kraftigt förbättrar noggrannheten och tillförlitligheten vid medicinsk informationssökning jämfört med baslinjemodeller. Valet av tröskelvärden för hämtningsrelevans och valet av olika LLM påverkar också RAG-systemens prestanda. I artikeln föreslås en robust utvärderingsmetod för RAG-system inom medicinsk kvalitetssäkring och grunden läggs för att utvidga denna metod till andra kunskapsintensiva domäner. Sådana tillförlitliga utvärderingar kommer att bidra till utvecklingen av mer effektiva och tillförlitliga medicinska kvalitetssäkringssystem, vilket gynnar både vårdgivare och patienter.

Nyckelord

medicinsk frågesvar, Stora språkmodeller, Retrieval-Augmented Generation, Utvärdering.

Acknowledgments

I would like to express my sincere gratitude to my thesis supervisor, Amirhossein Layegh Kheirabadi from KTH Royal Institute of Technology (KTH), whose expertise and thoughtful guidance have been crucial throughout this research journey.

I am deeply grateful to my sincere gratitude to my examiner, Amir H. Payberah, who continuously showed great interest in the progress of the project.

I want to give a special thanks to my friend Qihuang Xie for his collaborative spirit and insightful contributions, which helped shape many of the ideas and analyses presented here. It has been a rewarding and enlightening experience to work with him.

Lastly, I want to sincerely thank my family for their support and encouragement throughout my two years of studies in Sweden. Their devotion to me and their unfailing faith in me has been my biggest inspiration.

I am grateful to all of the individuals listed above for their combined assistance and encouragement in producing this thesis.

Stockholm, December 2024
Tangyujun Han

Contents

List of Figures

List of Tables

Listings

List of acronyms and abbreviations

AI	Artificial Intelligence
HyDE	Hypothetical Document Embeddings
KTH	KTH Royal Institute of Technology
LLM	Large Language Model
LSTM	Long Short-Term Memory
MAP	Mean Average Precision
MRR	Mean Reciprocal Rank
NLG	Natural Language Generation
NLP	Natural Language Processing
QA	Question Answering
RAG	Retreival-Augmented Generation
RNN	Recurrent Neural Networks

Chapter 1

Introduction

1.1 Background

Accessing reliable and personalized health information is a great challenge in today's digital healthcare. Traditional search methods, such as browsing medical websites and research papers, are often time-consuming and fail to ensure accurate or personalized information. Alternatively, consulting medical experts may not always be feasible, as doctors may not be available to answer questions when needed.

The above challenges have found a promising solution with the development of Large Language Models (LLMs). By utilizing advanced Natural Language Processings (NLPs) capabilities, LLMs can significantly enhance information retrieval and comprehension [1]. Such models can process and make sense of massive amounts of data and generate insightful answers. In healthcare, LLMs can address patients' concerns and provide medical knowledge assistance.

However, applying LLMs in real-life critical scenarios is dangerous due to their potential to produce hallucinations [2]. They may provide convincing but incorrect medical information and mislead patients. To address this problem, the Retreival-Augmented Generation (RAG) technique has been introduced to largely improve reliability by integrating information retrieved from authoritative medical sources.

1.2 Problem

While the development of LLM-based RAG applications shows promise in providing personalized and trustworthy medical information and answers to patients, the current absence of a standard benchmark for evaluating such RAG systems raises concerns.

Many existing Natural Language Generation (NLG) evaluation frameworks require much manual work for scoring [3]. While these evaluation frameworks might give a system's performance a quantitative assessment, such heavy human annotations can take a long time.

Some scholars have suggested using LLMs as the evaluator for assessment and defined some LLM-based metrics [4]. Although this method largely reduces the manual workload, its reliability remains questionable [5].

The research questions can be summarized as follows:

1. How should we evaluate a RAG system's performance, and what aspects and metrics should we consider?

2. Can we have a systematic method to evaluate the RAG systems in medical Question Answering (QA) tasks without relying on extra human annotations?

1.3 Purpose

This thesis aims to fulfill the urgent need for a standard method to evaluate RAG systems in medical QA tasks. Such a method will enable the convenient and accurate evaluation of related RAG systems, contributing to the development of more sophisticated RAG systems. This is important for the advancement in the field of health informatics and the realization of the transformational power of LLMs in patient education and engagement. In that way, researchers could be guaranteed a more efficient research performance, health professionals could provide improved service, and patients could get more dependable, personalized answers.

1.4 Goals

This project aims to propose a structured method of the evaluation of RAG systems in medical QA tasks that will drive the development and optimization of RAG systems in the medical field. The sub-goals are listed as follows:

1. Construct Different RAG Systems: Develop various systems such as LLM-only, Naive RAG, Hypothetical Document Embeddings (HyDE) RAG as test subjects.

2. Design Evaluation Metrics: Well-define the metrics that accurately assess the performance of RAG systems in medical QA scenarios.
3. Construct a Benchmark Dataset: Build a benchmark dataset for medical QA that can be utilized to test the performance of RAG systems.
4. Test and Analyze: Employ different configurations and conduct evaluations of different RAG systems by using the benchmark dataset and making an analysis.

1.5 Research Methodology

This research follows a positivist approach that uses statistical analysis and objective observation to evaluate how well RAG systems perform in medical QA tasks. Various RAG systems will be constructed and compared using a benchmark dataset with different language models. This study uses experimental research to systematically manipulate and measure various RAG configurations under controlled conditions. Quantitative analysis is performed to measure and compare different metrics, including accuracy, precision, recall, F1 score, Mean Average Precision (MAP), and Mean Reciprocal Rank (MRR).

1.6 Contributions

This thesis makes several significant contributions in medical RAG QA systems. The work proposes a novel, systematic evaluation method of medical RAG QA systems without relying on additional human annotation, which is the key gap in the current methodology. It defines evaluation metrics such as accuracy, precision, recall, F1 score, MAP, and MRR for effective measurement of both retrieval and generation aspects of RAG systems. This work finally examines the impact of various retrieval relevance thresholds and LLMs on the effectiveness of RAG systems. It gives valuable insights for optimizing retrieval settings and selecting the suitable model. These contributions advance the development of more effective and dependable medical QA systems that offer better services for healthcare providers and patients.

1.7 Delimitations

This project only focuses on the accuracy of Yes/No answers. This narrow scope might not fully represent the complexity of medical QA tasks, as it needs to consider other questions or whether the subsequent explanations fully align with the ground truth. Moreover, other potential metrics such as comprehensibility, relevance, and risk assessment of the generated answers should be considered as the evaluation focuses on retrieval. Furthermore, the study deals exclusively with evaluating RAG systems designed for medical information retrieval. Establishing similar benchmarks in other fields is still widely open and requires further efforts from the research community. Lastly, the project only uses OpenAI's GPT models, with no testing conducted on other LLMs. Future research should explore other LLMs, which may reveal different strengths and weaknesses within the RAG systems.

1.8 Ethics and sustainability

The project aims to help develop better RAG systems in medicine, which provide patients with real-time, helpful information and answers, helping them better understand and assess their physical condition. Given that the information deals with a person's health, the generated content has to be reliable, as any misleading or wrong information could have serious consequences for patient care.

Ethical guidelines have been strictly followed throughout this study, ensuring safeguarding concerning the integrity and reliability of the research process. The Benchmark dataset modified from widely used public datasets aligns with established data protection regulations.

Moreover, the potential biases inherent in LLMs and RAG systems were critically examined to ensure that the generated medical answers do not propagate misinformation or pose a threat to health. By leveraging the potential of LLMs, this thesis advocates for the advancement of public health informatics and exemplifies how the transformative power of LLMs can enhance patient education and engagement.

1.9 Structure of the thesis

The structure of the thesis is as follows: Chapter 2 presents the background of the proposed topic, including knowledge of LLM, RAG and related works about the evaluation of RAG system. The research methodology employed in this thesis is summarized in Chapter 3. Chapter 4 gives details on evaluation setting and RAG system design. Chapter 5 introduces the definition of the metrics we used and analyzes the performance of the evaluation. Finally, Chapter 6 summarizes the entire project and discusses potential directions for further study.

Chapter 2

Background

2.1 NLP

NLP is an important branch of Artificial Intelligence (AI) that focuses on implementing machines' ability to process, understand, and produce human language. The scope of work for this field ranges from language translation and semantic analysis to summarization of texts [6]. The primary goal of NLP is to program computers to comprehend the semantics and nuances of human language, an objective that continues to present challenges due to the inherent complexity of natural languages.

Recent advancements in NLP, driven by the transformer architecture, have revolutionized the capabilities of LLMs, greatly enhancing the quality of text generation. These state-of-art models show a superior ability to understand user queries while responding with contextually relevant responses [6] highlighting the transformative impact of NLP technology.

2.2 Transformer

The Transformer architecture, first presented by Vaswani et al. in 2017 [7], which relies on its self-attention mechanism to learn complex relationships and long-range dependencies within input sequences, revolutionized the world of NLP. From 2.1, it is evident that this architecture has two primary components: encoder and decoder. Each component consists of multiple stacked layers, incorporating multi-head self-attention mechanisms and feedforward neural networks. The encoder processes the input sequence into a fixed or continuous representation. Then, the decoder uses it to generate the output sequence.

The Transformer's self-attention mechanism enables each word in one

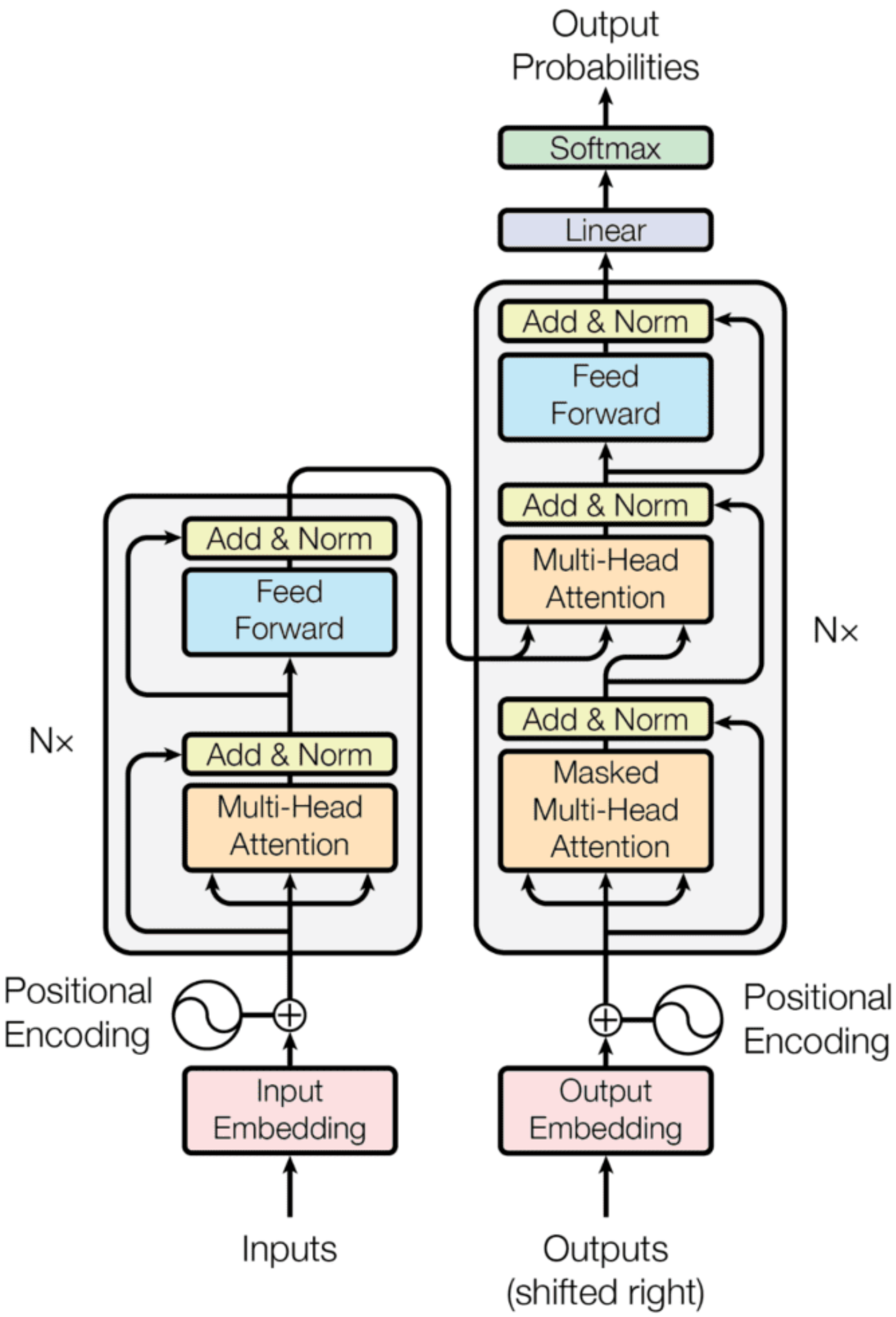


Figure 2.1: The architecture of the Transformer [7]

sequence to pay attention to all other words in that same sequence with different degrees of importance based on context, which is crucial for understanding and generating language. This is done by mapping every token to a collection of values, keys, and queries. It involves the calculation of dot products to get scores and the application of the softmax function to yield the attention weights. The model architecture uses multiple-heads in parallel to capture diverse aspects of word relationships. Moreover, positional encoding is combined with the input embeddings to convey information about the order of tokens since maintaining sequence structure requires that. By this design, the Transformer can perform most of the NLP tasks, including QA, language translation, summarization[7].

In contrast to Recurrent Neural Networkss (RNNs) and Long Short-Term Memorys (LSTMs), which process inputs sequentially, the Transformer takes an entire sequence as input in parallel, significantly improving computational efficiency and handling extensive contextual information. This parallel processing ability and the model's capacity to concentrate on relevant parts of the input using a self-attention mechanism make transformers more efficient and scalable compared to earlier neural network architectures. As a result, models like GPT [8], BERT [9], and T5 [10], which are based on Transformers, have become the backbone of many state-of-the-art LLMs, revolutionizing the field of NLP with their superior performance and flexibility.

2.3 LLM

LLMs have emerged as a groundbreaking advancement in the research of NLP and AI. With enormous sizes of parameters and complex model architectures, these models have attained unparalleled performance across diverse language-related activities such as text processing, generation, translation, and QA [11]. The growing interest in LLMs is because they can learn deep patterns and semantic structures from huge text corpora, enabling them to demonstrate human-level fluency and contextual understanding while performing language processing tasks.

In practice, LLMs are utilized across various domains. For example, in healthcare, they may analyze data for insight into diseases or serve as agents offering personalized support in customer service. Even as they have such broad applicability, developing these models remains a complex and time-consuming task, requiring extensive computational resources and expertise.

One of the main properties of LLMs is that they can learn from enormous volumes of text data through pre-training on extensive corpora, driven by the exponential growth of data. In this pre-training, the models use the context that previous words provide to predict the following word in a given sequence. This enables the models to learn subtle patterns and semantic relationships in language. Additionally, by incorporating small amounts of high-quality labeled data, these models can be fine-tuned to enhance performance in particular user-focused domains and achieve practical levels of utility.

From a model perspective, LLMs leverages a large order of magnitude parameter to store extensive knowledge, leading to the emergence of new capabilities. The use of prompt-based methods unifies natural language understanding and generation tasks, providing a more intuitive human-computer interface. The transformer architecture at the heart of LLMs has proven extremely good at acquiring contextual information and long-range dependencies within sequences of tokens. These features guarantee that the LLM represents a milestone in the field of NLP.

ChatGPT

GPTs represent a set of deep learning models proposed by OpenAI using Transformer architecture. These models have become critical in generative AI applications, including the well-known chatbot ChatGPT.

In the last five years, a significant development in LLMs has achieved remarkable results on various tasks.

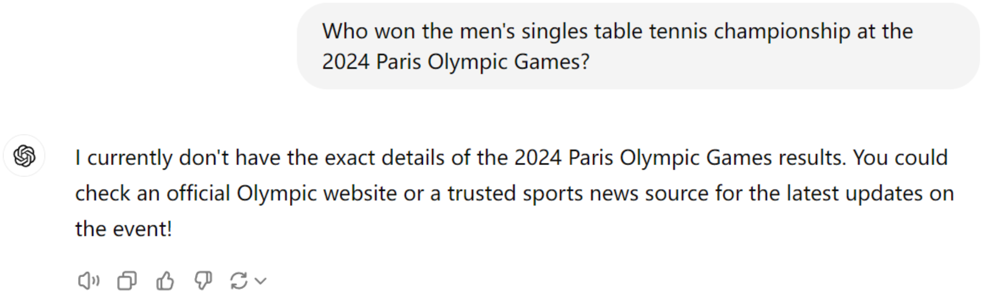
Before 2017, most NLP models relied on supervised learning and, therefore, could only perform tasks that they had been explicitly trained on [12]. The appearance of the Transformer architecture in 2017 [13] provided the basis for semi-supervised learning techniques that enabled the development of BERT [9] and GPT [8]. These models were based on unsupervised pre-training followed by supervised fine-tuning and thus can support multitasking functionality.

GPT models have evolved quickly, with each one improving over the previous model by a great margin. GPT-1 came first in 2018 [8], introducing semi-supervised training that first relied on unsupervised pre-training followed by supervision in fine-tuning and attained surprising results in several tasks within natural language processing. Then, in 2019, GPT-2 was developed based on its predecessor with 1.5 billion parameters to produce excellent scores in a variety of NLP benchmarks [14]. Released in 2020, GPT-3 [15] provided a quantitative leap with its 175 billion parameters and performed well on various tasks, thanks to its massive and diverse training dataset.

ChatGPT from OpenAI was the following revolutionary tool in conversational AI in 2022. It showed outstanding skills in language understanding, text generation, and knowledge-based reasoning [16]. Two months after its release, ChatGPT reached a record milestone of 100 million active users, becoming the fastest-growing consumer application in history [17]. The incredible speed of this success has attracted lots of attention, including governments, industry leaders, and academia, driving a new wave of AI races and offering enormous opportunities to practically all fields.

Another giant leap came in March 2023 with GPT-4. GPT-4 is more reliable and creative, adding the possibility of multimodal input and increasing its utility farther than ever before [18]. This model is expected to impact fields such as healthcare and medical research greatly. The development of ChatGPT marks a milestone in addressing core challenges in NLP, representing a critical step toward the realization of general AI with the potential to transform numerous fields and industries.

2.4 RAG



Who won the men's singles table tennis championship at the 2024 Paris Olympic Games?

I currently don't have the exact details of the 2024 Paris Olympic Games results. You could check an official Olympic website or a trusted sports news source for the latest updates on the event!

Figure 2.2: An example of the limitation

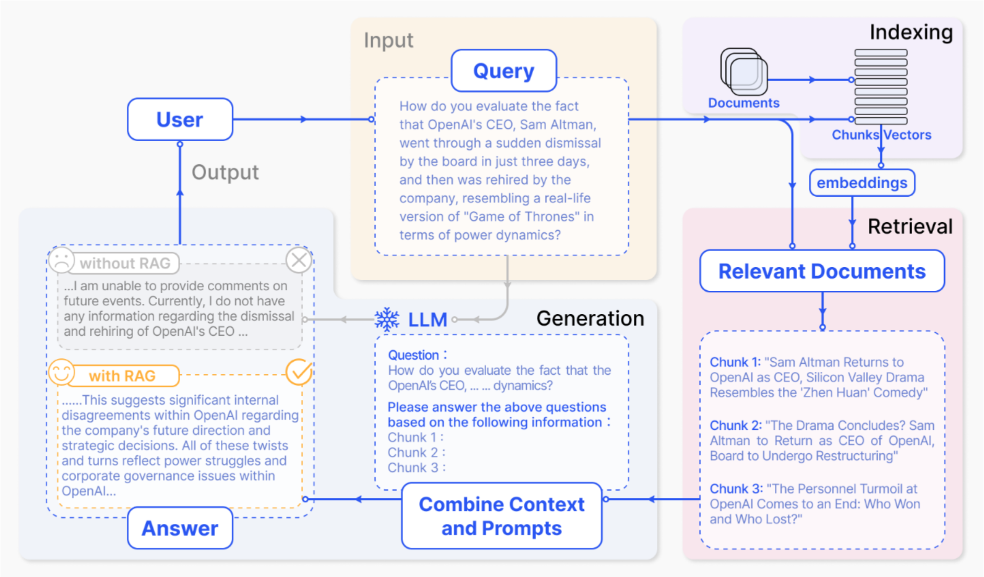
LLMs have demonstrated exceptional reasoning skills across many topics. However, they are not perfect and have some limitations. Fig 2.2 shows an example. In this case, the model is unable to provide the result when asked about the recent men's singles table tennis championship at the 2024 Paris Olympic Games. This example demonstrates the model's reliance on pre-existing knowledge up to a specific cutoff date and inability to access real-time data. This limitation becomes problematic when dealing with tasks requiring access to private data or information introduced after the model's training period.

Additionally, LLMs are prone to hallucinations [2], where sometimes the responses of these models sound very good but are based on fake or incorrect information. This is of most concern in complex and knowledge-based tasks because it is hard to determine whether the given response is correct.

To address these limitations, new techniques have been invented to overcome shortcomings and enhance the precision and dependability of the generative AI models. RAG [19] incorporates a component for information retrieval into the text generation model. In the RAG system, relevant documents are retrieved from a given source using the input. These documents are combined with the initial input prompt to generate the final output.

The RAG methods succeed in a lot of NLP tasks, including relation extraction [20], machine translation [21], and dialogue [22]. By explicitly acquiring prior external knowledge, RAG allows for greater flexibility and improved learning through analogy. Moreover, RAG can be fine-tuned through various strategies to achieve optimal performance, and its internal knowledge can be updated efficiently without retraining the entire model.

Process



How do you evaluate the fact that OpenAI's CEO, Sam Altman, went through a sudden dismissal by the board in just three days, and then was rehired by the company, resembling a real-life version of "Game of Thrones" in terms of power dynamics?

...I am unable to provide comments on future events. Currently, I do not have any information regarding the dismissal and rehiring of OpenAI's CEO ...

......This suggests significant internal disagreements within OpenAI regarding the company's future direction and strategic decisions. All of these twists and turns reflect power struggles and corporate governance issues within OpenAI...

Question :
How do you evaluate the fact that the OpenAI's CEO, ... dynamics?
Please answer the above questions based on the following information :
Chunk 1 :
Chunk 2 :
Chunk 3 :

Chunk 1: "Sam Altman Returns to OpenAI as CEO, Silicon Valley Drama Resembles the 'Zhen Huan' Comedy"

Chunk 2: "The Drama Concludes? Sam Altman to Return as CEO of OpenAI, Board to Undergo Restructuring"

Chunk 3: "The Personnel Turmoil at OpenAI Comes to an End: Who Won and Who Lost?"

Figure 2.3: An example of the RAG process applied to QA [23]

A representative application of RAG used for QA is illustrated in Figure 2.3. It mainly involves three main stages: indexing, retrieval, and generation.

In the indexing phase, documents are loaded and split into small, manageable chunks. This segmentation allows efficient indexing and enables LLMs to handle data within their token limitations. After that, an embedding model transforms the chunks into vector representations, which are saved in a vector database. This process typically happens offline so that data can be prepared for quick retrieval during real-time queries.

After the data has been indexed, the retrieval phase starts. When a user submits a query, the retriever component searches the Vector Store for relevant chunks based on similarity scores. The chunks retrieved along with the user query are given as input to LLM.

For the generation phase, the final contextually relevant responses are generated by the LLM using a prompt that includes both the user's query and the retrieved contexts. Such an iterative process of retrieval followed by generation forms the backbone of RAG application, enabling it to effectively address user queries by integrating information from the indexed database.

2.5 Prompt Engineering

Prompt engineering is an emerging field that comes together with the rise of LLMs. It is the art and science of writing an effective prompt. This text string contains natural language directions to improve the behavior of LLMs, giving them a particular style or manner [24]. In contrast to methods like fine-tuning, which irreversibly changes the model's behavior, prompt engineering, on the one hand, enables users to temporarily influence the behavior of a model simply by rewording or structuring a request differently. This feature will allow users to tailor outputs for particular objectives or tasks in real time without touching the underlying model.

A good prompt clarifies the expectations and gives the model adequate context that best suits the need for which a response is being made. Several prompting strategies are commonly employed, including zero-shot, one-shot, and few-shot learning. These approaches differ in the number of examples provided to the model alongside the task instructions: zero-shot learning involves supplying only the instructions without any examples, one-shot learning includes a single example, and few-shot learning incorporates a small number of examples. These examples serve to guide the model in generating the desired output. The design of these prompts is a critical task of the whole process, as specific prompt designs are more effectively interpreted by models, leading to improved accuracy and quality in their responses[25]. Due to the probabilistic nature of LLMs, crafting optimal prompts requires iterative testing and refinement.

While the field remains in its early stages, ongoing research continues to explore new methodologies to improve the design and effectiveness of prompts. This study area is key to maximizing the potentials of LLMs in various applications.

2.6 Langchain

LangChain [26] is an open-source development framework specifically created to support the creation of LLM-based applications in Python and TypeScript. A core component of this framework is called chain, which serves as a fundamental building block, allowing developers to connect prompts and responses modularly. This component makes complex workflows convenient by linking chains with different functions together.

In the context of RAG applications, LangChain facilitates application development by providing a whole package of utilities, making it easier to preprocess and augment data. These include Document Loaders, Text Splitters, Embedding Models, Vector Stores, and Retrievers. LangChain puts all of these components into an overall framework, allowing large models to easily connect with external knowledge and handle queries on specialized topics beyond the model's training data.

2.6.1 Chunking

Long documents typically need to be broken down into smaller segments that fit within the model's context window. Although this seems very simple, it can actually involve lots of complexity since semantic integrity needs to be preserved. We need to ensure that related pieces of information stay together to maintain coherence and context within the text.

Different document types, such as PDFs, Python scripts, or Markdown, require different methods tailored to their unique structure. LangChain has a complete suite of document transformers catering to each type. We can then load and manipulate the documents using LangChain effectively.

Once the appropriate document loaders have been chosen, one important thing to consider is the chunk size. This now brings us to how the split process works. 2.4 illustrate this mechanism. While processing the document, text will accumulate in a chunk. When the chunk has reached the target size, it is marked as a distinct piece of text. Then, the process starts again, creating a new chunk with some overlap from the previous segment. This overlap will

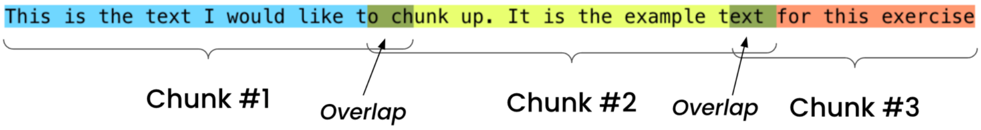
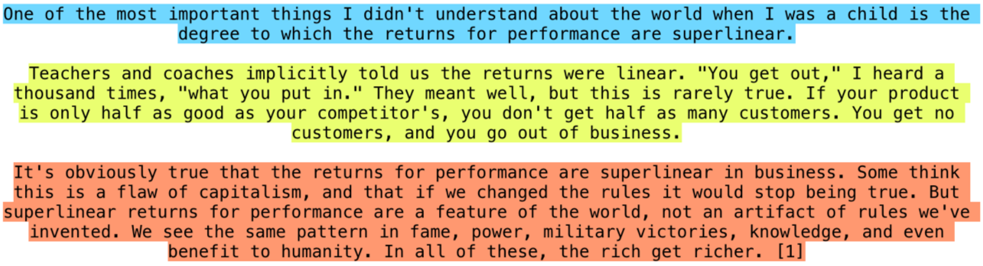

This is the text I would like to chunk up. It is the example text for this exercise
Chunk #1
Overlap
Chunk #2
Overlap
Chunk #3

Figure 2.4: Naive text splitter

help in the continuity and coherence of chunks, maintaining comprehensible segmented text, which can be further processed or analyzed with meaning.



One of the most important things I didn't understand about the world when I was a child is the degree to which the returns for performance are superlinear.

Teachers and coaches implicitly told us the returns were linear. "You get out," I heard a thousand times, "what you put in." They meant well, but this is rarely true. If your product is only half as good as your competitor's, you don't get half as many customers. You get no customers, and you go out of business.

It's obviously true that the returns for performance are superlinear in business. Some think this is a flaw of capitalism, and that if we changed the rules it would stop being true. But superlinear returns for performance are a feature of the world, not an artifact of rules we've invented. We see the same pattern in fame, power, military victories, knowledge, and even benefit to humanity. In all of these, the rich get richer. [1]

Figure 2.5: Recursive character text splitter [27]

The recursive character text splitter is the default and most common text splitter in Langchain. Its advantage is that it allows splitting documents more intelligently by considering several separators in a hierarchical manner.

As it can be seen in 2.5, the recursive character text splitter tries to split the document using the most significant separators first, such as double new lines ("\n\n"), which typically indicate paragraph breaks. If further splitting is necessary, it will proceed to use single new lines ("\n"), then spaces (" "), and finally, individual characters if needed.

This approach helps the recursive character text splitter maintain the document structure better than a simple character count-based split. In this way, the divisions of the text could be preserved logically and serve various applications more effectively.

2.6.2 Embeddings

Embeddings are numerical vector representations that encode textual data into a multidimensional space, capturing the text's semantic meaning and contextual relationships. These vectors allow for advanced text analysis and manipulations such as semantic search, whereby one can identify and compare

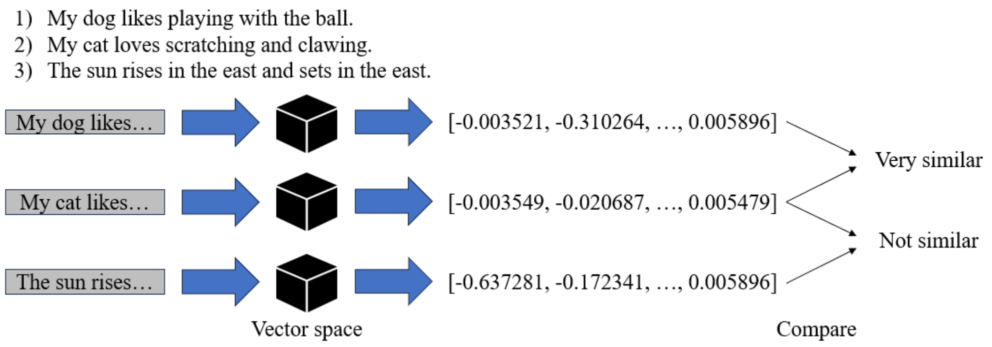


Figure 2.6: Three examples of embeddings

pieces of text based on their semantic similarities using vector similarity scores such as cosine similarity.

Figure 2.6 depicts this process of embedding various sentences into the vector space, which finally converts them into high-dimensional vector representations. The semantic relatedness of sentences can be determined by calculating the similarity between these vectors. Sentences about animals and their activities may yield similar vectors because of their semantic contents, whereas a sentence about a natural phenomenon may give a dissimilar vector. This process is critical for effecting necessary transformations in many natural language processing tasks since it allows for more complexity in understanding the textual content.

Specialized machine learning models, known as embedding models, generate these vector representations from textual input. Companies like OpenAI, Cohere, and Hugging Face provide such models. Each embedding model creates vectors that capture the semantic meaning of input text so that tasks like similarity searches, clustering, and other forms of semantic analyses can be completed.

LangChain unifies all these embedding models by providing a convenient interface for multiple providers, making it easy to use different embedding models.

2.6.3 Vector Store

Vector Stores are a specialized class of data storage systems designed to manage and query large volumes of unstructured data by leveraging embeddings. These systems are optimized for similarity search and efficient retrieval of semantically relevant information. Notable examples of Vector

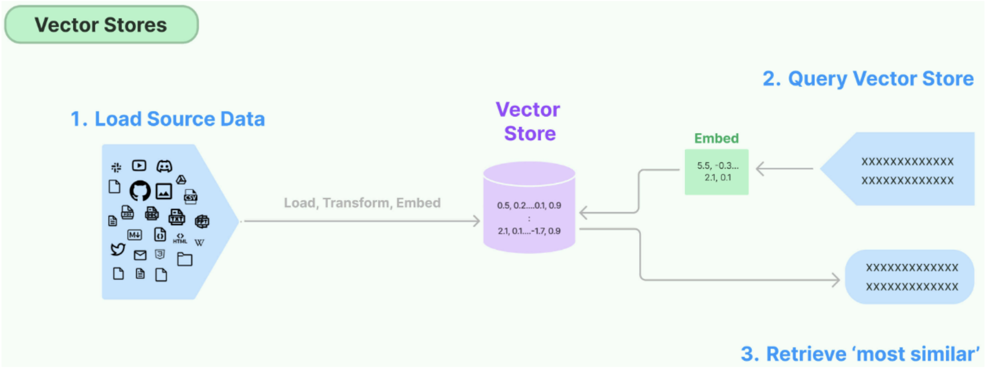


Figure 2.7: Vector Store

Stores include Chroma, Pinecone, and FAISS (Facebook AI Similarity Search), each offering unique features to support high-performance retrieval in various applications.

LangChain played an important role in integrating these capabilities of the Vector Stores. It provides a streamlined framework for building applications that harness the power of Vector Stores. LangChain simplifies the implementation of loading and processing documents, generating embeddings, and conducting vector searches, providing a robust and scalable framework for managing and querying unstructured data efficiently.

2.6.4 Retrievers

During query time, the unstructured query gets projected to a vector representation that the retriever uses to fetch the most similar vectors in the Vector Store. A retriever is an abstraction layer that returns documents for these unstructured queries.

Retriever takes a string query as input and produces an output with a list of relevant documents. This provides flexibility in document retrieval according to query needs. By integrating Vector Stores, retrievers can leverage the power of embeddings, enhancing their capability to conduct semantic searches and identify the most pertinent documents within large datasets. LangChain plays an important part in this ecosystem by providing a standardized interface through which many embedding models and retrievers can easily integrate, optimizing the retrieval process.

2.7 Evaluation

While the prospects for RAG systems look promising, a standard method is needed to evaluate the performance.

Traditional human evaluation methods rely on human raters to score the queries, answers, and retrieved contexts based on relevance and quality. Although such methods can ensure high accuracy, they are labor-intensive and time-consuming, limiting scalability and reusability.

Using LLMs as evaluators can be an attractive alternative to address these challenges [28, 29]. LLMs can judge the quality of the integration between retrieval and generation components using criteria such as relevance, coherence, and informativeness and make the process much quicker and scalable.

RAGAS is a framework [4] for evaluating the RAG systems. It defines a wide range of metrics covering different dimensions of the performance of systems without requiring extensive human annotations. RAGAS is particularly useful in allowing quick evaluation cycles. By constantly monitoring and assessing different aspects of the RAG pipeline, developers can find opportunities for enhancement, implement effective retrieval strategies, and refine prompts crafting. This systematic evaluation is critical to advance the effectiveness of RAG systems for providing relevant, coherent, and helpful responses to users.

However, there are still questions about the precision and reliability of the LLMs as evaluators, for they may have embedded biases that could affect the grade on generated answers [5]. Besides, their sensitivity to input variations [30] also raises concerns.

2.8 Related work

LLMs have been studied in medicine to develop and enhance clinical decision-making and assist in performing medical QA tasks.

Almanac [31] integrates the language models with medical guideline retrieval and treatment recommendations. It reports significant improvements in factuality, completeness, and safety across various specialties after evaluating 130 clinical scenarios by five physicians.

Similarly, Self-BioRAG [32] introduces a specialized framework for the biomedical text that focuses on producing explanations, retrieving documents specific to a given domain, and reflecting on the generated answers. Trained on 84,000 filtered biomedical instruction sets, it emphasizes the importance of specific components like tailored retrievers and specialized corpora.

BEEP [33] presents a novel approach for predicting clinical outcomes by integrating patient-specific medical literature into predictive models. By analyzing individual clinical notes, BEEP retrieves relevant research papers to enhance predictions, achieving significant improvements compared to baseline models.

However, existing evaluations are not comprehensive. They often focus on the quality of the generated answer without assessing the performance of the retrieval component. Moreover, the reliance on substantial human annotations makes repeated evaluations troublesome. Our study bridges the gap of systematic evaluations of RAG systems in medicine QA.

2.9 Summary

This chapter reviews key advances in NLP, including how Transformer-based LLMs like GPT have changed the landscape of language understanding and generation. It discusses the limitations of LLMs, including their reliance on pre-existing knowledge and tendency to produce hallucinations. It introduces RAG as a solution that integrates information retrieval with text generation to enhance accuracy and reliability.

Additionally, the chapter looks at related works like Almanac, Self-BioRAG, and BEEP, each showing the potential to combine LLMs with retrieval mechanisms. It concludes by identifying a gap in systematic evaluations of RAG systems in medical QA tasks, which the present study aims to address.

Chapter 3

Methods

This chapter outlined the research methodology used in this project. Section 3.1 describes the research process. Section 3.2 explains the research paradigm in detail. Section 3.3 concentrates on the techniques of data collection adopted in this study. Section 3.4 explains the experimental design. Section 3.5 evaluates the validity and dependability of the method and the data collected. Section 3.6 explains how the data analysis was done. Finally, Section 3.7 explains the framework selected to assess this project.

3.1 Research Process

This section outlines the key steps and methodologies to solve the research problems defined in 1.2.

3.1.1 Understanding the problem domain

The detailed understanding of the problem domain helped to start the research work for formulating the problem statement. It included studying related literature and delving deep into the existing landscape regarding the evaluation of RAG systems. Such a preliminary step required insights into the reliability, efficiency, and performance that needed to be guaranteed in evaluating the RAG systems.

3.1.2 Defining research goals

This project aimed to fulfill the urgent need for a convenient method to evaluate RAG systems in medical QA tasks, which shaped the research goals. We categorized these goals into phases and sub-goals, such as constructing various RAG systems, developing evaluation metrics, building a benchmark dataset, and testing.

3.1.3 Design and Conduct Experiment

The research methodology was designed and conducted with suitable experiments to validate the proposed evaluation method. It included pre-defining relevant metrics for evaluation, writing relevant code, gathering data to ensure reliability, and then processing it.

The conduct phase focused on systematically executing the experiments, monitoring for consistency, adjusting parameters as needed, and ensuring that all variables were controlled effectively to maintain the integrity of the results. The process was also documented in this stage so that it became reproducible.

3.1.4 Discussion and Results

The final stage was the discussion of results. At this stage, the importance of our results had to be assessed regarding their limitations or influencing factors, and then a conclusion based on the chosen evaluation criteria had to be drawn. This last stage was crucial for evaluating the project's overall success and provided valuable insights into the system's limitations.

3.2 Research Paradigm

The research paradigm primarily follows positivism, focusing on collecting numerical data and employing quantitative methods such as statistical analysis for interpretation. The primary aim is to acquire knowledge through observation and objective analysis. Data collection follows a deductive approach, utilizing a predetermined set of metrics for measurement and analysis.

3.3 Data Collection

We mainly collected the retrieved contexts and answers from the response during each query process in the RAG system. The collection is easily achievable if we define the response's structure and content using LangChain. The collected data is then used to calculate various metrics for subsequent analysis.

Sampling

Due to economic reasons, we sampled some data from the complete benchmark dataset to use as the test set. This sample adheres to the distribution of the ground truth contexts in the original dataset. The sample size is 114 to maintain the distribution of the ground truth contexts consistent with the original dataset.

3.4 Experimental Design

To test the proposed approach for evaluating the performance of RAG systems in medical QA, we constructed various RAG systems: Naive RAG, HyDE RAG, and LLM-only. To accurately assess their performance, we defined a set of comprehensive evaluation metrics tailored to the retrieval and generation domain. These metrics included accuracy, precision, recall, F1 score, MAP and MRR).

A diverse and representative dataset of medical questions and answers was built to serve as the benchmark for testing the RAG systems. The constructed RAG systems were subjected to rigorous testing using this benchmark dataset. The evaluation process involved deploying each RAG system in a controlled environment to ensure consistent testing conditions. A series of question-answer sessions were conducted where each system processed the same set of medical questions.

Based on their responses, selected metrics were calculated for each system. The results were analyzed to identify the strengths and weaknesses of each system, with specific attention given to areas where the systems underperformed to pinpoint potential areas for improvement. We also explored various models and relevance score thresholds to study their effect on system performance.

Using the above experimental design allows us to give a thorough and reliable evaluation of the various RAG systems that aim to demonstrate the dependability and efficiency of our proposed approach to evaluate the RAG systems.

Test environment

This project is implemented entirely through code. We use Python (version 3.10) as the primary programming language for the testing environment.

LangChain is the development framework for the LLM-based medical QA system, leveraging the OpenAI 'GPT-3.5-turbo-0125' and 'GPT-4o-mini' model. The temperature for the model has been kept at 0 to ensure consistency of the model outputs to a high degree. This temperature setting is critical when dealing with sensitive medical information. While these higher temperatures allow the model to produce more creative and diverse outputs, our need for accuracy and reliability necessitates constraining the model to a lower temperature setting. All these prerequisites set a sound basis for the development, testing, and performance of the evaluation of our RAG systems to keep their focus on producing accurate and relevant medical information.

3.5 Assessing reliability and validity of the data collected

This section evaluates the validity and dependability of the method and the data collected.

3.5.1 Validity of method

To ensure the validity of our method, we processed the same set of medical questions through each of the RAG systems so that any comparisons would be fair. Moreover, because of our extended testing and systematic analysis, the strengths and weaknesses of every RAG system became apparent, pointing to areas for improvement. Such measures ensured that our approach would be both valid and robust for the actual performance of RAG systems in the medical QA domain.

3.5.2 Reliability of method

We had a reliable experimental design because the test settings were consistent in a controlled environment, and the QA sessions are standardized to allow repeatability. Since each RAG system was deployed in the same controlled environment, no external variable might interfere with its performance, ensuring that only the capability of the systems is evaluated. Standardized QA sessions resulted in each system processing the same set of medical questions under the same conditions. Such an approach would allow us to repeat the experiment more than once and expect to achieve the same results, thereby reinforcing the reliability of our measurements and conclusions.

3.5.3 Data validity

The validity of the collected data was ensured through a modified representative dataset of medical questions and answers reflecting real-world scenarios. We adopted standard and widely accepted metrics for evaluation, including accuracy, precision, recall, F1 score, MAP, and MRR, which comprehensively captured different aspects of performance. These measures ensured that the data was accurate and that conclusions could reflect the systems' true capabilities.

3.5.4 Reliability of data

To ensure the reliability of our data, each of the RAG systems was evaluated multiple times under identical configurations. Repeating identical conditions offered consistency in the values measured, confirming reliability. This rigorous and systematic process thus ensured reliability and reproducibility in results regardless of the number of runs.

3.6 Planned Data Analysis

3.6.1 Data Analysis Technique

We used standard quantitative metrics to compare this study's various RAG systems. For each RAG system, we computed these metrics and averaged the values resulting from multiple runs by ensuring the reliability of the outcome. After that, the average of each metric is compared across various systems to perform the analysis.

3.6.2 Software Tools

This thesis was implemented with Pycharm and Jupyter Notebook in a Conda environment. Libraries used in the experiment were

- **Langchain:** Open-source framework to build LLM applications.
- **Faiss:** Open-source library for efficient similarity search and clustering of dense vectors.
- **Matplotlib:** Famous Python plotting library provides static, animated, and interactive visualizations to generate plots and graphs.

- **Pandas:** Powerful Python library applied to data analysis and manipulation. It offers data structures like DataFrames that facilitate effective work with structured data.

3.7 Evaluation framework

In the initial phase, we evaluated the function of the prototype by verifying that the responses contained the correct and expected formatted data. This initial check ensured that the data returned by the system included both the retrieved contexts and the corresponding answers accurately.

After verifying the prototype's basic functionality, we proceeded to a more detailed evaluation by testing different RAG systems, including Naive RAG, HyDE RAG, and LLM-only. Each system was assessed under the same conditions using our benchmark dataset. We employed a collection of evaluation metrics tailored to the retrieval and generation domain and conducted multiple tests to evaluate each system's performance.

Chapter 4

Implementation

4.1 Evaluation Setting

This study's main objective is to assess RAG systems in a manner that reflects patients' real-life medical information requirements. Consequently, our evaluation includes the following settings:

- Retrieval Utilization: Answering real-world medical questions is inherently challenging due to their knowledge-intensive nature. Retrieval must be utilized during the evaluation process to ensure that the generated answers are reliable and well-founded.

- Zero-Shot Learning: Given that real-world medical questions are frequently asked without prior similar examples, in our setting, the RAG systems should be assessed in a zero-shot context where the use of few-shot learning is not allowed

- Question-Only Retrieval: In realistic settings, QA only knows the question. The retrieval should only use the question as the initial input. This approach offers a more authentic evaluation for RAG systems.

- Yes/No QA Evaluation: Yes/No QA is one of the easiest and most common practical methods to evaluate on a large scale [34]. One can directly check the generated answers without annotations.

These methodological choices collectively provide a comprehensive and systematic method to evaluate the RAG systems in QA tasks without relying on extra human annotations.

4.2 Metrics

An important research question is what aspects and metrics we should consider to evaluate a RAG system's performance. Given the architecture of RAG systems introduced in 2.4, it is natural to assess their performance across two key dimensions: Retrieval and Generation.

4.2.1 Accuracy

Accuracy is a critical metric in evaluating the performance of the generation component of a RAG system, as it directly reflects the system's ability to produce correct and reliable answers.

In the prompt, we have already forced that the answers returned by LLM begin with "Yes" or "No." This format makes it possible to compare the results directly to the golden standard answers, starting with "Yes" or "No." If the responses match, the accuracy of the QA instance will be 1. In case of disagreement, the recorded accuracy will be 0.

The accuracy of the QA instance can be computed using the formula that follows:

$$\text{Accuracy} = \frac{\sum_{i=1}^{N} \delta(\text{response}_i, \text{golden standard}_i)}{N}$$

where:

- N is the total number of QA instances.
- response_i is the answer generated by LLM for the i-th instance.
- golden standard_i is the correct (golden standard) answer for the i-th instance.
- $\delta(x, y)$ represents the Kronecker delta function, which is 1 if $x = y$ and 0 otherwise.

4.2.2 Precision

Precision is a measure of the accuracy of the positive contexts retrieved by RAG systems. It represents the fraction of true positive contexts retrieved among all the contexts retrieved. High precision is essential as it reflects the retrieval system's ability to minimize irrelevant information, ensuring the relevance of the retrieved contexts.

For precision, the formula is:

$$\text{Precision} = \frac{TP}{TP + FP}$$

where:

- TP (True Positives) is the number of positive instances correctly retrieved by the RAG systems.
- FP (False Positives) is the number of false instances incorrectly retrieved by the RAG systems.

4.2.3 Recall

Recall measures the ability of RAG systems to identify all relevant instances, reflecting the proportion of true positive contexts among the total actual positives. High recall is crucial in ensuring that the system retrieves as many relevant contexts as possible, minimizing the risk of missing critical information. It reveals the system's effectiveness in covering the breadth of relevant data.

We use the following formula to calculate Recall:

$$\text{Recall} = \frac{TP}{TP + FN}$$

where:

- TP (True Positives) is the number of positive instances correctly retrieved by the RAG systems.
- FN (False Negatives) is the number of positive instances the RAG systems failed to retrieve.

4.2.4 F1 score

The F1 score is the harmonic mean of Precision and Recall, providing a balanced measure that accounts for both false positives and false negatives. It plays an important role when there is an uneven class distribution or when both precision and recall are crucial. In RAG systems, a high F1 score indicates a well-rounded performance, revealing the system's ability to retrieve relevant contexts accurately while minimizing irrelevant ones. This balance is critical in applications like medical QA, where both comprehensive coverage and precision are necessary to ensure reliable and accurate information retrieval.

The formula for F1 score is:

$$\text{F1 score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

4.2.5 MAP

MAP measures the quality of a ranked list of retrieved results. It calculates the mean of Average Precision scores for each query, considering the order of results. This metric is vital as it evaluates whether relevant contexts are retrieved and their ranking. A high MAP score indicates that relevant information is consistently placed higher in the retrieval list, enhancing the system's efficiency by presenting the most valuable contexts early. For RAG systems, this reveals the system's capability to prioritize highly relevant contexts.

The formula for MAP is:

$$\text{MAP} = \frac{1}{Q} \sum_{q=1}^{Q} \text{AP}(q)$$

where:

- Q is the total number of queries.
- $\text{AP}(q)$ is the Average Precision for query q, calculated as:

$$\text{AP}(q) = \frac{\sum_{k=1}^{n} P(k) \times \text{rel}(k)}{\text{number of relevant documents}}$$

- $P(k)$ is the precision at position k, calculated as:

$$P(k) = \frac{\text{The number of relevant documents in the top } k \text{ retrieved documents}}{k}$$

- $\text{rel}(k)$ is an indicator function that is 1 if the retrieved document at rank k is relevant and 0 otherwise.

4.2.6 MRR

MRR is a metric that measures the effectiveness of a system in retrieving relevant documents. It considers the rank of the first relevant retrieved result. It ensures effectiveness by highlighting the system's ability to present relevant information quickly, reflecting how efficiently systems can retrieve valuable content. For RAG systems, a high MRR score indicates that the system consistently retrieves relevant contexts early in the ranked list.

The formula for MRR is:

$$\text{MRR} = \frac{1}{Q} \sum_{q=1}^{Q} \frac{1}{\text{rank}_q}$$

where:

- Q is the total number of queries.
- rank_q is the rank position of the first relevant retrieved document for query q.

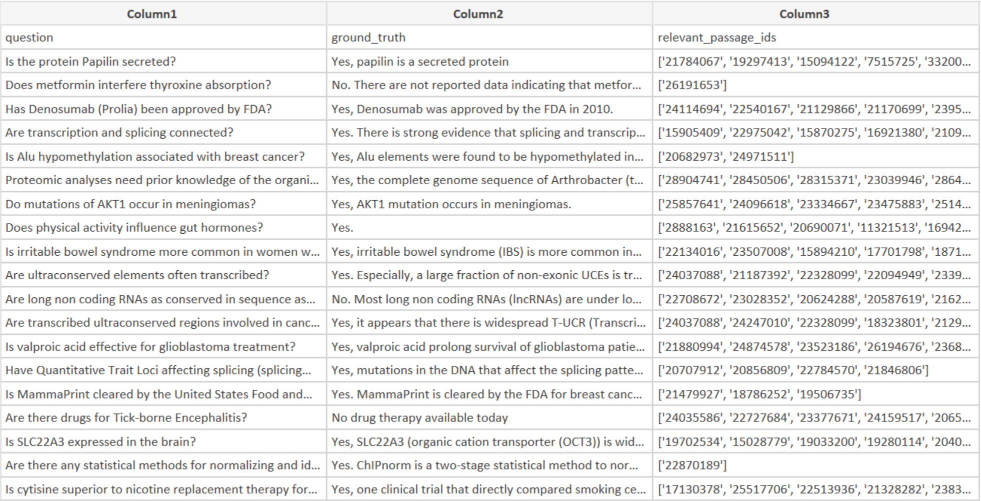
4.3 Benchmark Dataset

BioASQ [35] is a competition focusing on large-scale biomedical semantic indexing and QA. It seeks to assess how well systems can use data from biomedical articles to semantically index large volumes of scientific papers and provide accurate, easily comprehensible responses to queries in natural language.

BioASQ Task B aims to gather accurate and understandable responses to biomedical research questions. The English questions given to the systems taking part in Task B are designed by biomedical professionals to represent the information requirements of everyday life. The systems must return relevant articles, snippets from the corpus, and English summaries. These questions, gold standard reference answers, and associated documentation are included in the BioASQ-QA dataset created for Task B, providing a realistic and challenging benchmark for developing and evaluating biomedical QA systems.

In order to align with the evaluation settings discussed earlier, we selected Yes/No questions from the BioASQ-QA dataset. To facilitate a more accurate assessment of retrieval performance, we removed a portion of the questions with an excessive number of relevant passages. This filtered dataset, named

BioASQ-QA-Y/N, consists of 776 questions, allowing us to focus on a more tractable and reliable evaluation of RAG systems.



Column1	Column2	Column3
question	ground_truth	relevant_passage_ids
Is the protein Papilin secreted?	Yes, papilin is a secreted protein	['21784067', '19297413', '15094122', '7515725', '33200...
Does metformin interfere thyroxine absorption?	No. There are not reported data indicating that metfor...	['26191653']
Has Denosumab (Prolia) been approved by FDA?	Yes, Denosumab was approved by the FDA in 2010.	['24114694', '22540167', '21129866', '21170699', '2395...
Are transcription and splicing connected?	Yes. There is strong evidence that splicing and transcrip...	['15905409', '22975042', '15870275', '16921380', '2109...
Is Alu hypomethylation associated with breast cancer?	Yes, Alu elements were found to be hypomethylated in...	['20682973', '24971511']
Proteomic analyses need prior knowledge of the organi...	Yes, the complete genome sequence of Arthrobacter (t...	['28904741', '28450506', '28315371', '23039946', '2864...
Do mutations of AKT1 occur in meningiomas?	Yes, AKT1 mutation occurs in meningiomas.	['25857641', '24096618', '23334667', '23475883', '2514...
Does physical activity influence gut hormones?	Yes.	['2888163', '21615652', '20690071', '11321513', '16942...
Is irritable bowel syndrome more common in women w...	Yes, irritable bowel syndrome (IBS) is more common in...	['22134016', '23507008', '15894210', '17701798', '1871...
Are ultraconserved elements often transcribed?	Yes. Especially, a large fraction of non-exonic UCEs is tr...	['24037088', '21187392', '22328099', '22094949', '2339...
Are long non coding RNAs as conserved in sequence as...	No. Most long non coding RNAs (lncRNAs) are under lo...	['22708672', '23028352', '20624288', '20587619', '2162...
Are transcribed ultraconserved regions involved in canc...	Yes, it appears that there is widespread T-UCR (Transcri...	['24037088', '24247010', '22328099', '18323801', '2129...
Is valproic acid effective for glioblastoma treatment?	Yes, valproic acid prolong survival of glioblastoma patie...	['21880994', '24874578', '23523186', '26194676', '2368...
Have Quantitative Trait Loci affecting splicing (splicing...	Yes, mutations in the DNA that affect the splicing patte...	['20707912', '20856809', '22784570', '21846806']
Is MammaPrint cleared by the United States Food and...	Yes. MammaPrint is cleared by the FDA for breast canc...	['21479927', '18786252', '19506735']
Are there drugs for Tick-borne Encephalitis?	No drug therapy available today	['24035586', '22727684', '23377671', '24159517', '2065...
Is SLC22A3 expressed in the brain?	Yes, SLC22A3 (organic cation transporter (OCT3)) is wid...	['19702534', '15028779', '19033200', '19280114', '2040...
Are there any statistical methods for normalizing and id...	Yes. ChIPnorm is a two-stage statistical method to nor...	['22870189']
Is cytisine superior to nicotine replacement therapy for...	Yes, one clinical trial that directly compared smoking ce...	['17130378', '25517706', '22513936', '21328282', '2383...

Figure 4.1: A demonstrative screenshot of BioASQ-QA-Y/N dataset

Figure 4.1 provides an example of the BioASQ-QA-Y/N dataset. In this dataset, the columns include the following: the question column, which contains the original biomedical Yes/No questions provided by experts; the ground truth column, which contains the gold standard answers, providing ”Yes” or ”No” responses along with brief explanations for each; and the relevant passage ids column, which lists the IDs of relevant passages from the corpus, indicating the specific passages that the system should retrieve from the Vector Store during evaluation. The relevant passage IDs are essential as they specify the exact passages to be retrieved, enabling a more targeted assessment of retrieval performance.

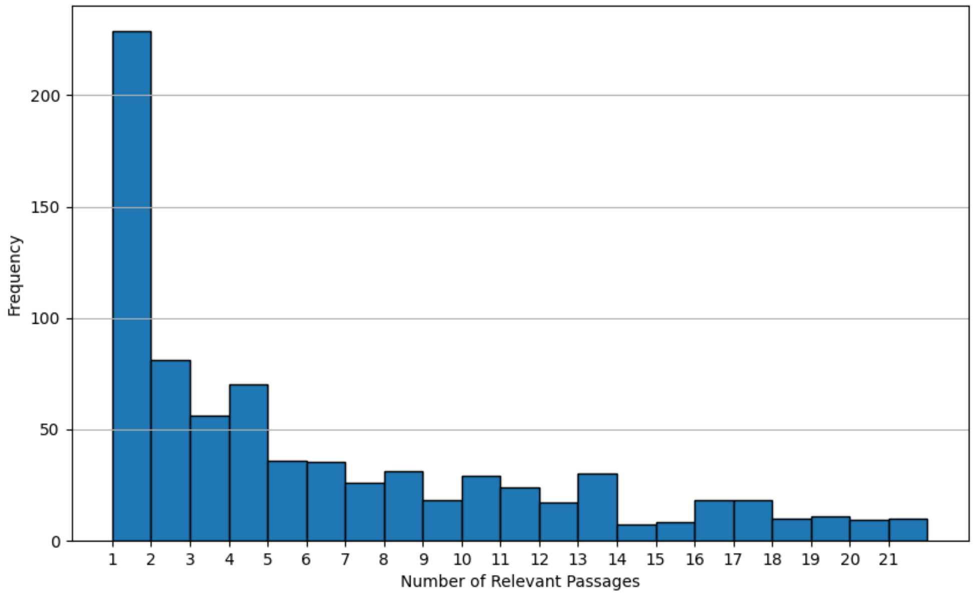


Figure 4.2: Distribution of Number of Relevant Passages in BioASQ-QA-Y/N

Figure 4.2 illustrates the distribution of the number of relevant passages within this dataset. Our sampled dataset will also follow this distribution.

In this project, we will use this BioASQ-QA-Y/N dataset as the benchmark dataset to evaluate different RAG systems.

4.4 RAG Systems Design

4.4.1 Parameter selection

Parameter	**Values**
Data loader	CSVLoader
Vector stores	Chroma / FAISS
Models	OpenAI ’GPT-3.5-turbo-0125’ / ’GPT-4o-mini’
Embedding Model	OpenAI Embeddings
Relevance score	Cosine similarity

Table 4.1: Experiment parameters used in the project

Table 4.1 summarizes the key parameters and components used in the experimental setup. For data processing, the system employs the CSVLoader to handle data input. It utilizes two types of Vector Stores, Chroma and FAISS, for efficient similarity search operations. The OpenAI 'GPT-3.5-turbo-0125' model is the main engine driving the computations, while OpenAI Embeddings convert input data into vector representations. The system applies Cosine similarity as the relevance score metric to measure the similarity between embeddings, ensuring an accurate comparison of vector closeness in the embedding space.

4.4.2 LLM-only

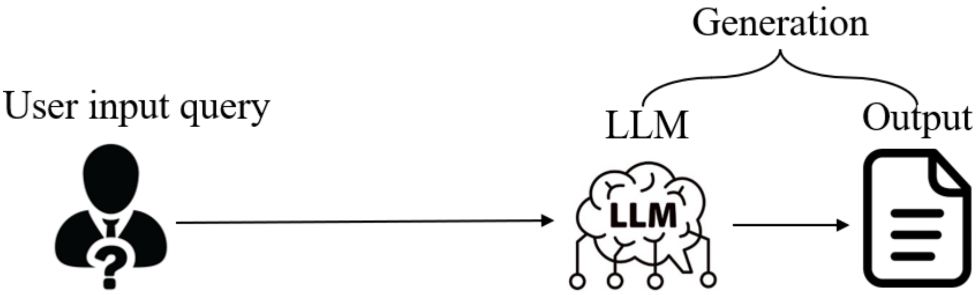


Figure 4.3: Workflow of LLM-only system.

In this project, LLM-only system serves as a baseline. As is shown in Fig4.3, it does not incorporate any document retrieval. Instead, it relies entirely on the generation component to answer questions based on pre-existing knowledge within the LLM. Fig 4.4 shows an example of a query of LLM-only system. In this system, we only consider accuracy regarding the evaluation metric.

Question: Are artificial blood cells available?
Ground truth answer: No, the critical point for the breakthrough for artificial blood products has not come yet but could be ahead. **Generated Answer:** No, artificial blood cells are not currently available for use in medical practice. Research is ongoing to develop artificial blood cells that can potentially be used in the future to treat various medical conditions. However, at this time, traditional blood transfusions using donated blood remain the standard treatment for patients in need of blood.
Metrics: **Accuracy:** 1

Figure 4.4: An instance of a query of LLM-only system

4.4.3 Naive RAG

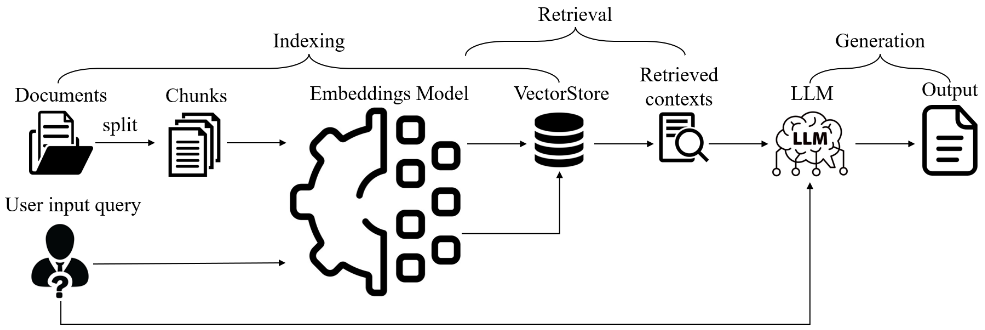


Figure 4.5: Workflow of Naive RAG system.

A Naive RAG system employs a straightforward approach to combining retrieval and generation techniques to answer queries. The retrieval component uses simple and direct methods, specifically basic vector similarity searches based on the index of the query using the embedding model, with the relevance score calculated by cosine similarity. The system searches through the predefined Vector Store, generated from the corpus, to find those most relevant to the user's query. This method is designed to quickly identify documents that contain terms related to the query without employing complex algorithms or deep learning models. Once the relevant documents are retrieved, the generation component takes over. It processes the content of these documents to understand the context and extract useful information that can be used to generate an answer using NLP techniques. The system uses a prompt to guide the generation process, ensuring the extracted information is formulated into a coherent and informative response.

Question:
Are artificial blood cells available?

Relevant contexts id: ['28406466', '20234994']
Retrieved Contexts:
context 1: Formerly developed resuscitation fluids solely imitated the main function of the blood -oxygen transport. A research driven by the army requested an oxygen carrier that does not need cross typing and cooled storage. ...
context id: '20234994'

Ground truth answer: No, the critical point for the breakthrough for artificial blood products has not come yet but could be ahead.
Generated Answer: No, artificial blood cells are not widely available. Research and development in this area have faced challenges, with many products failing to achieve convincing study results. Some ventures are currently investigating third and fourth generation products of artificial blood components, but a breakthrough has not yet been achieved.

Metrics:
Accuracy: 1
Precision: 1
Recall: 0.5
F1 score: 0.667
MAP: 1
MRR: 1

Figure 4.6: An instance of a query of Naive RAG system

Fig 4.6 presents a real example. In the case of RAG system, we have the list of relevant contexts id, which contains the ideal contexts the RAG system is expected to retrieve. Additionally, the actual retrieved contexts are available. These data enable the computation of metrics like accuracy, recall, F1 score, MAP, and MRR, which quantify the retrieval performance, ensuring a comprehensive assessment of the system's effectiveness in retrieving information.

4.4.3.1 Prompt

```
You are a helpful medical expert, and your task is to answer
a Yes/No medical question using the retrieved contexts. If
the contexts are not provided, answer the question directly
based on your medical knowledge. Your answer must be faithful
to the question.

The answer must start with either ”Yes” or ”No”, followed by
a explanation related to the question. Use three sentences
maximum and keep the answer concise.

question:{question}
context: {contexts}
```

Listing 4.1: Prompt of RAG

Listing 4.1 presents the prompt utilized by the RAG system to generate answers for yes/no medical questions. The prompt explicitly instructs the model to assume the role of a medical expert, aiming to provide an informed response based on the retrieved contexts. In cases where no relevant contexts are retrieved, the model generates an answer based on its prior medical knowledge. The prompt specifies that the response must begin with either ”Yes” or ”No,” followed by a brief explanation that provides context or justification, restricted to a maximum of three sentences. This consistent answer format makes the calculation of accuracy convenient.

4.4.3.2 Code

```
def naive_chain(retriever, llm_name):
    llm = ChatOpenAI(model_name=llm_name, temperature=0)
    prompt_rag = ChatPromptTemplate.from_template(
   rag_template)
    final_rag_chain = (
            RunnableParallel(
            {”contexts”: itemgetter(”question”) | retriever |
    format_docs,
             ”question”: itemgetter(”question”)}
            ) |
            {”answer”: prompt_rag | llm | StrOutputParser(),
            ”contexts”: itemgetter(”contexts”)}
    )
    return final_rag_chain
```

Listing 4.2: code of Naive RAG chain

Listing 4.2 presents the implementation of the Naive RAG chain, developed using the Langchain framework. The function integrates the retriever and a LLM into the RAG process. The ChatOpenAI component is initialized with a specified language model, and a zero temperature is used to ensure deterministic output. The chain proceeds by constructing a prompt template using ChatPromptTemplate. Subsequently, the RunnableParallel component facilitates the parallel execution of retrieval and generation tasks. The language model prompts the question and the retrieved contexts to generate the final answer. This implementation exemplifies a streamlined approach to building a RAG system.

4.4.4 Hyde RAG

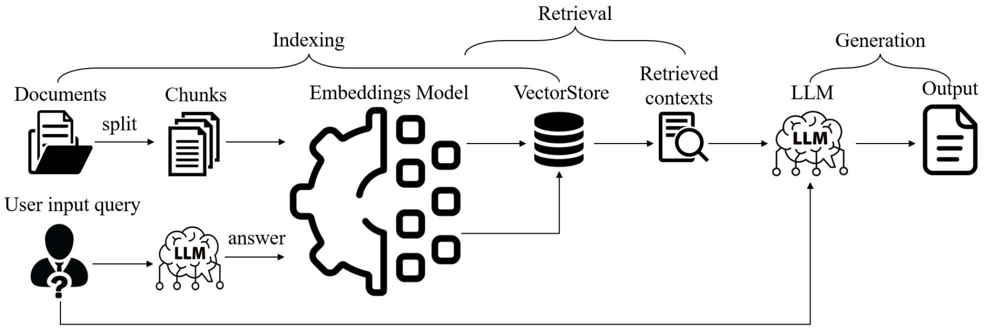


Figure 4.7: Workflow of HyDE RAG system

HyDE [36] is a new technique intended to improve the performance and reliability of information retrieval systems. Traditional embedding retrievers often struggle to generalize effectively across new domains, a challenge HyDE aims to address.

Fig 4.7 shows the workflow of HyDE RAG system. It operates by initially generating a hypothetical document based on the initial query. This process involves zero-shot prompting a language model to create a synthetic document that encapsulates the relevant patterns of the initial query. Once the hypothetical document is generated, it is encoded into an embedding vector using the embedding model. This embedding subsequently identifies similar documents within the document embedding space based on vector similarity. The primary advantage of this approach is that the generated hypothetical document may be closer to the ground truth context in the vector space than the initial query, which enhances the accuracy of identifying relevant contexts.

Since the search results are sourced directly from a trusted knowledge base, the risk of returning inaccurate responses by the language model is minimized, making HyDE particularly useful in sensitive applications such as healthcare, where precise and reliable information is crucial. By refining search mechanisms and reducing the time spent filtering out irrelevant documents, HyDE enhances overall productivity.

Question:
Are artificial blood cells available?

Hypothetical Document:
Artificial blood cells, also known as synthetic red blood cells or blood substitutes, are a promising area of research in the field of biotechnology and medicine. These artificial blood cells are designed to mimic the functions of natural red blood cells, such as carrying oxygen and nutrients throughout the body. …

Relevant Contexts id: ['28406466', '20234994']
Retrieved Contexts:
context 1: Formerly developed resuscitation fluids solely imitated the main function of the blood -oxygen transport. A research driven by the army requested an oxygen carrier that does not need cross typing and cooled storage. …
Context id: '20234994'
Context 2: Recombinant human erythropoietin (rhEpo, epoetin) has proved beneficial in preventing transfusion-dependent anaemia in patients with chronic kidney disease. Apart from copied epoetins distributed in less regulated markets, …
Context id: '24748667'

Ground truth answer: No, the critical point for the breakthrough for artificial blood products has not come yet but could be ahead.
Generated Answer: No, artificial blood cells are not widely available for clinical use. Despite decades of research and development, many products have not achieved convincing study results or FDA approval. Some ventures are currently investigating third and fourth generation products, but a breakthrough in artificial blood products has not yet occurred.

Metrics:
Accuracy: 1
Precision: 0.5
Recall: 0.5
F1 score: 0.5
MAP: 1
MRR: 1

Figure 4.8: An instance of a query of HyDE RAG system

Figure 4.8 illustrates an example of a query processed by the HyDE RAG system. In this instance, the language model first generates a hypothetical document, providing a synthesized, contextually relevant response to the question. This hypothetical document connects the initial query and the retrieved contexts, guiding the retrieval component towards more accurate and relevant information. By creating a preliminary summary of potential answers, the system can refine the retrieval process, thereby improving the overall quality and relevance of the retrieved information.

4.4.4.1 Prompt

```
Please write a scientific passage to answer the question.

question:{question}
Passage:
```

Listing 4.3: prompt of HyDE

Listing 4.3 shows the prompt used by the HyDE system for generating a hypothetical document. The prompt requests the model to write a scientific passage in response to the provided question.

4.4.4.2 Code

```
def hyde_chain(retriever, llm_name=):
     prompt_hyde = ChatPromptTemplate.from_template(
    hyde_template)
     llm = ChatOpenAI(model_name=llm_name, temperature=0)
     generate_docs_for_retrieval = (
             prompt_hyde | llm | StrOutputParser()
     )
     retrieval_chain = generate_docs_for_retrieval | retriever
     prompt_rag = ChatPromptTemplate.from_template(
    rag_template)
     final_rag_chain = (
             RunnableParallel(
             {"contexts": retrieval_chain | format_docs,
             "question": itemgetter("question")}
             )|
             {"answer": prompt_rag | llm | StrOutputParser(),
             "contexts": itemgetter("contexts")}
     )
     return final_rag_chain
```

Listing 4.4: code of HyDE chain

Listing 4.4 presents the implementation of the HyDE chain. Unlike the Naive RAG chain, the function adds the step of generating a hypothetical document prior to retrieval. Initially, ChatPromptTemplate creates a HyDE prompt, which is passed to the LLM to generate this hypothetical document. This document serves as an intermediary, guiding the retrieval of relevant contexts. The final response is then produced using the recovered contexts and a RAG prompt.

4.5 Embedding and Retriever setting

We use OpenAI's embedding model for the entire process in this project. For the LangChain retriever, the retrieved documents are ranked based on their similarity scores in descending order. It then usually returns the results in two ways: one returns the top k highest-scoring results, and the other returns only the results with scores above a specified threshold. The code of these two retrievers is shown in the List 4.5.

```
retriever = vectorstore.as_retriever(search_kwargs={”k”: 4})
retriever = vectorstore.as_retriever(
    search_type=”similarity_score_threshold”,
    search_kwargs={”score_threshold”: 0.75}
)
```

Listing 4.5: two kinds of retriever

From Fig 4.2, we can see that the number of relevant passages (ground truth contexts) varies significantly. Using the top k retrieval method is not suitable for this dataset. Therefore, we chose the second approach, filtering the results whose scores exceed a specified threshold.

The choice of this threshold becomes particularly important. If the value is too low, the retriever may include some potentially irrelevant contexts in the returned results, reducing the retrieval efficiency. On the other hand, if the value is set too high, we might miss some crucial contexts. Both scenarios can impact the generation process.

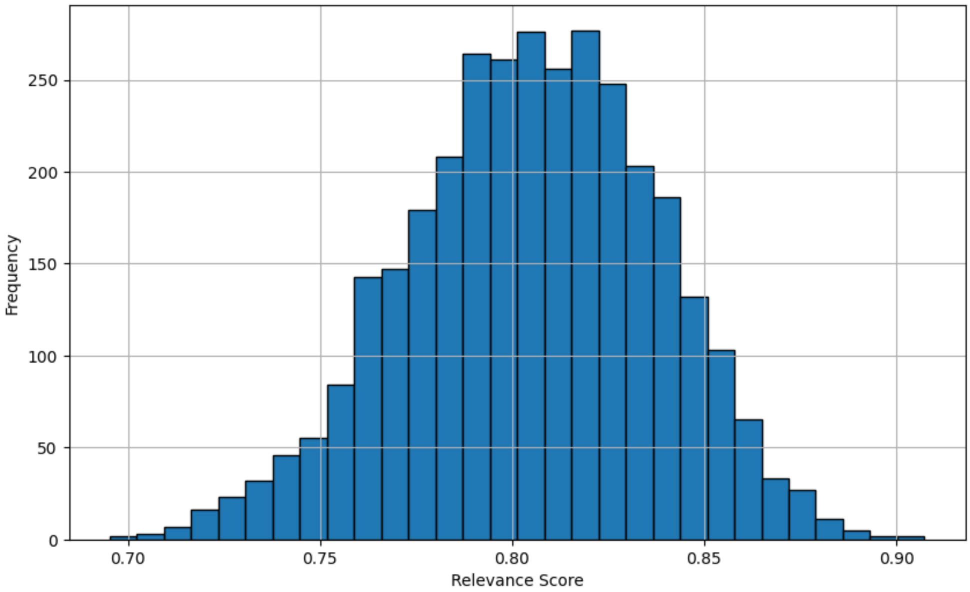


Figure 4.9: Distribution of relevance scores of matched contexts

Percentile	**Relevance score**
10th percentile	0.762
25th percentile	0.783
50th percentile (median)	0.806
75th percentile	0.828
90th percentile	0.846

Table 4.2: Distribution percentiles of relevance score of matched contexts

Using the benchmark dataset, we first evaluated the Naive RAG system. The retriever used the top 21 (the maximum number of relevant passages) retrieval method. We then recorded the relevance scores of the retrieved documents and filtered out those that matched the ground truth contexts. Fig 4.9 shows the distribution of the relevance scores of those matched contexts. Moreover, Table 4.2 gives distribution percentiles of relevance score. We decided to use 0.762 and 0.783 as testing thresholds for our retriever to cover most ground truth contexts.

4.6 Variable controlling

Table 4.3: Independent Variables

Independent Variable	**Values**
Different QA systems	No RAG
	Naive RAG
	HyDE RAG
Models	GPT-3.5
	GPT-4o-mini
Relevance Score Threshold	0.762
	0.783

Table 4.4: Dependent Variables

Dependent Variable	**Values**
Measurement Metric	Accuracy
	Precision
	Recall
	F1 score
	MAP
	MRR

In summary, the experiment evaluates three QA systems—LLM-only, Naive RAG, and HyDE RAG, using two different models for the test, with relevance score thresholds of 0.762 and 0.783 applied. Metrics including accuracy, precision, recall, F1 score, MAP, and MRR are used to evaluate their performance. This structured approach enables a rigorous evaluation of each system's effectiveness, providing a comprehensive analysis of their capabilities in handling the medical QA tasks.

Chapter 5

Results and Analysis

5.1 Results

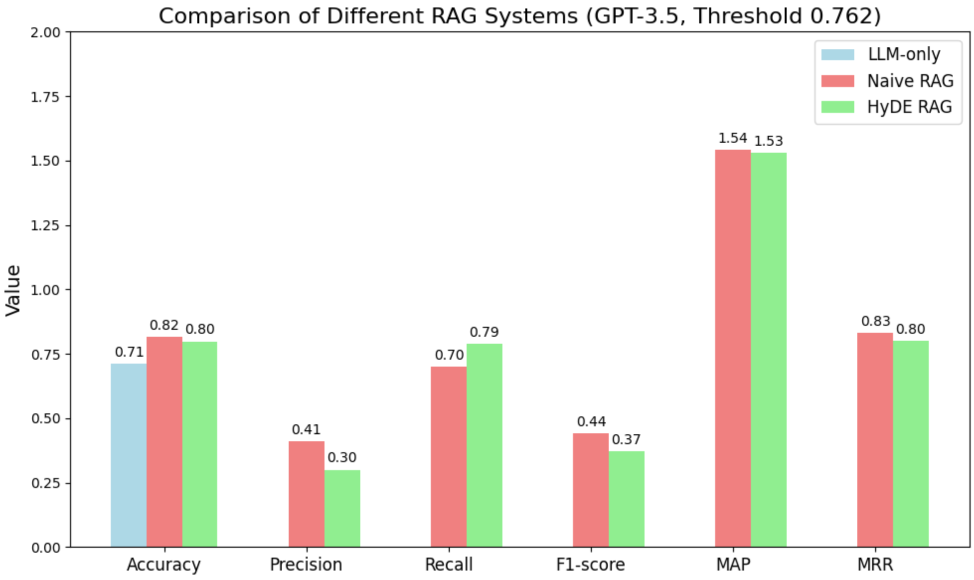


Figure 5.1: Comparison of different RAG systems using proposed metrics with 0.762 relevance score threshold and GPT-3.5 model

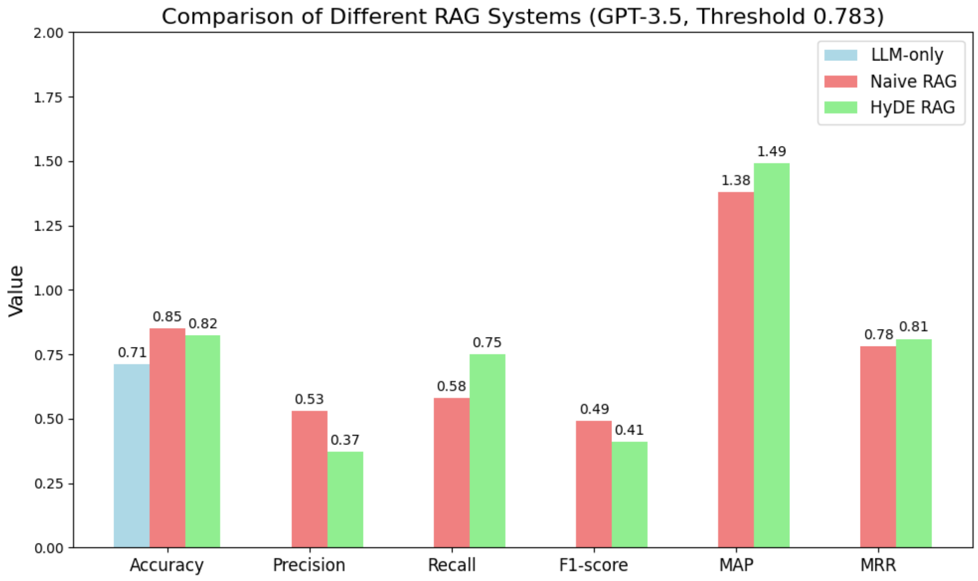


Figure 5.2: Comparison of different RAG systems using proposed metrics with 0.783 relevance score threshold and GPT-3.5 model

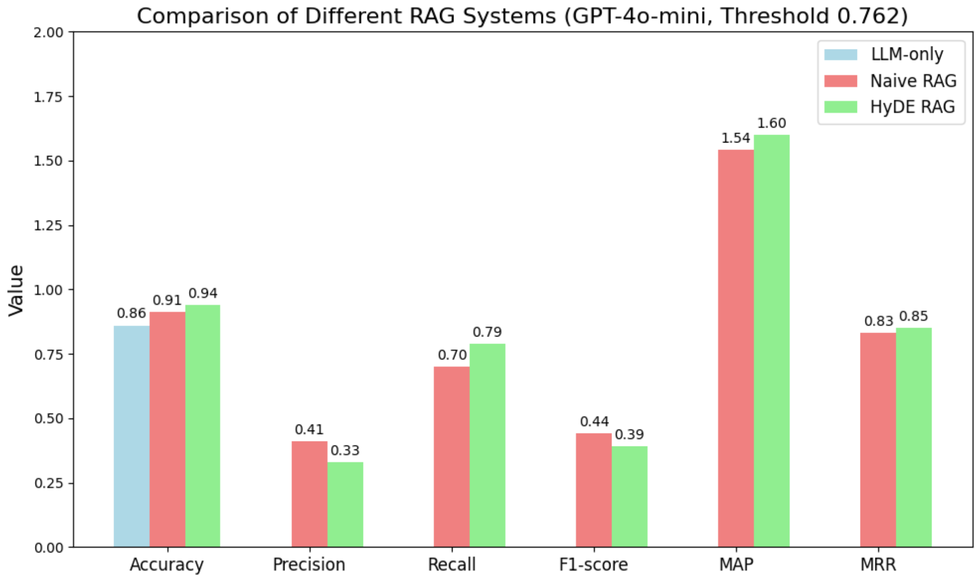


Figure 5.3: Comparison of different RAG systems using proposed metrics with 0.762 relevance score threshold and GPT-4o-mini model

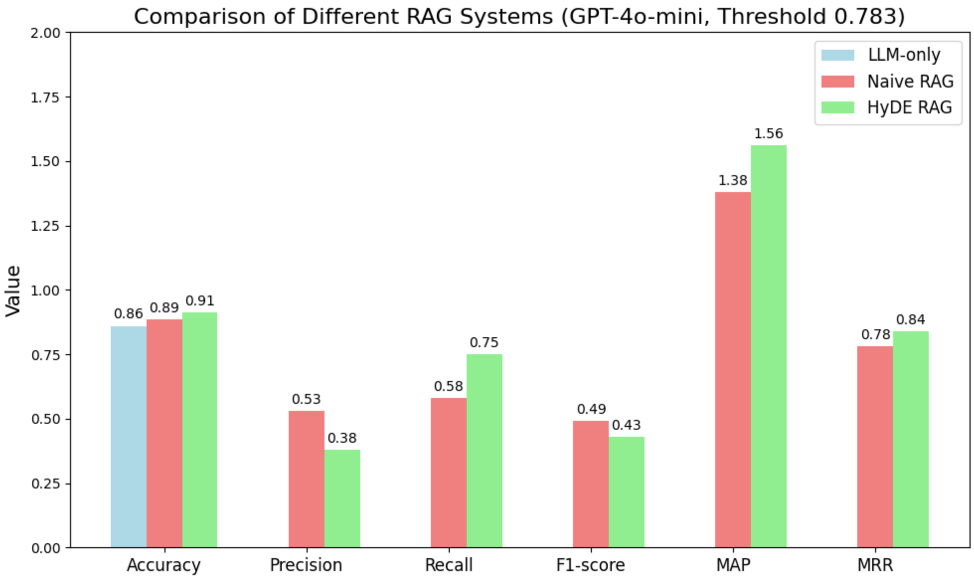


Figure 5.4: Comparison of different RAG systems using proposed metrics with 0.783 relevance score threshold and GPT-4o-mi model

5.2 Analysis

5.2.1 System Analysis

Across all configurations, LLM-only, Naive RAG, and HyDE RAG demonstrated varying levels of performance, influenced by both model type and threshold setting.

5.2.1.1 Accuracy

Overall, the RAG (Naive RAG and HyDE RAG) systems outperform the LLM-only system in terms of accuracy across both model types and thresholds. The highest accuracy is observed in the HyDE RAG system under the GPT-4o-mini model at a relevance score threshold of 0.762, achieving an accurate value of 93.86%. The result of accuracy suggests that incorporating retrieval mechanisms enhances the reliability of generated answers, contributing to more dependable outcomes in medical QA tasks. The result aligns with the existing research [33] where using RAG also helps improve the accuracy.

5.2.1.2 Precision

Naive RAG shows higher precision compared to HyDE RAG under all conditions. Precision means the accuracy of the positive contexts retrieved by RAG system. This difference can be attributed to the mechanism used by HyDE, which generates hypothetical texts based on the input question, subsequently retrieving similar documents. This approach increases the risk of introducing a portion of irrelevant information (FP) compared to question-only retrieval.

5.2.1.3 Recall

Recall values indicate that HyDE RAG generally outperformed Naive RAG across all configurations. Recall quantifies the RAG systems' capacity to identify all relevant instances. Although HyDE leads to a decrease in precision, this approach can also potentially uncover relevant documents (TP) that might have been missed by using question-only retrieval, making the recall score higher.

5.2.1.4 F1 score

F1 score reflects the balance between retrieving all relevant information (recall) and the correctness of those retrieved results (precision). Overall, Naive RAG demonstrates a better balance in retrieval performance, effectively managing the trade-offs in leveraging different retrieval strategies within RAG systems.

5.2.1.5 MAP

MAP values reveal that HyDE RAG consistently matches or exceeds the performance of Naive RAG. This indicates HyDE RAG's superior ability to rank relevant information higher in the retrieved list, which is crucial for the efficient and accurate provision of relevant medical information in QA tasks. Compared to the similarity between the question and the ground truth context, the generated hypothetical document is often more similar to the ground truth, which is why it can rank relevant information higher.

5.2.1.6 MRR

In terms of MRR, there is no significant difference between the HyDE RAG and Naive RAG systems, with both demonstrating similar performance. The relatively high MRR values indicate that both systems effectively rank the correct answers near the top.

5.2.2 Model Analysis

5.2.2.1 Accuracy

Regarding accuracy, the GPT-4o-mini model consistently outperforms GPT-3.5 across all systems and thresholds. Specifically, HyDE RAG achieves the highest accuracy with GPT-4o-mini at a threshold of 0.762, reaching 93.86%. This suggests that the more advanced model performs better in medical QA, effectively utilizing the retrieved information to generate more accurate and reliable answers.

5.2.2.2 Other Metrics

For the other metrics (Precision, Recall, F1 score, MAP, and MRR), the performance of the GPT-3.5 and GPT-4o-mini models is identical for the Naive RAG system, while there are minor differences observed for the HyDE RAG system. The reason for these differences is that the model only contributes to the generation phase, including the creation of hypothetical documents for HyDE RAG system, while not directly affecting the retrieval process. Consequently, any variation in the metrics for HyDE RAG is due to differences in how hypothetical content is generated rather than the retrieval mechanism itself.

5.2.3 Relevance Threshold Analysis

The analysis of relevance thresholds at 0.762 and 0.783 reveals distinct effects on the performance of the RAG systems across different metrics, highlighting the influence of this parameter in optimizing the retrieval process.

5.2.3.1 Accuracy

Across both relevance thresholds, the accuracy results show a significant variation for both Naive RAG and HyDE RAG systems. For the GPT-4o-mini model, accuracy is higher at the lower relevance threshold of 0.762 for both Naive RAG (91.23%) and HyDE RAG (93.86%), compared to the 0.783 threshold (Naive RAG: 88.60%, HyDE RAG: 91.23%). However, for the GPT-3.5 model, the results are the opposite. Accuracy is higher at the higher relevance threshold of 0.783 for both Naive RAG (85.09%) and HyDE RAG (82.46%), compared to the 0.762 threshold (Naive RAG: 81.58%, HyDE RAG: 79.82%).

This suggests that the relevance threshold's effect is influenced by various factors, and the choice of threshold should involve experimentation and a case-by-case analysis. While a lower threshold benefits the GPT-4o-mini model by providing more diverse retrievals that improve accuracy, the GPT-3.5 model sees improved performance with a higher threshold, likely due to stricter filtering leading to higher-quality context inputs.

5.2.3.2 Precision

With increases in threshold values from 0.762 to 0.783, we observe a corresponding increase in precisions across both models and systems. The increase in the threshold causes the system to avoid more documents whose relevance scores come below the threshold. This reduces the proportion of false positives (FP). As a result, the RAG system is more likely to retrieve the most relevant contexts, which will therefore improve precision.

5.2.3.3 Recall

Increasing the threshold from 0.762 to 0.783 results in a remarkable reduction in recall across all configurations. Since increasing the threshold reduces the number of documents returned, irrelevant contexts are filtered out, reducing true positives (TP). As fewer relevant instances are retrieved, the overall recall metric decreases, showing that a system is losing its capability to capture all possible relevant information.

5.2.3.4 F1 Score

Even though the increased threshold from 0.762 to 0.783 increases precision with a lowered recall, the overall F1 Score improves. This suggests that a higher threshold value optimizes the precision-recall tradeoff, improving overall retrieval performance. The better F1 score means that, at this point, the system can achieve more effective retrieval by better balancing the compromise between retrieving relevant documents and not retrieving irrelevant ones.

5.2.3.5 MAP

The values for MAP follow a slight drop for an increased threshold from 0.762 to 0.783. This is easily explainable - the more conservative threshold reduces the overall number of contexts returned, including some that probably would have had a lower relevance score but may still be useful. As fewer contexts are included in the retrieved set, the overall ranking of relevant contexts becomes less optimal, with a smaller MAP as a result.

5.2.3.6 MRR

In terms of MRR, no great difference is observed between the thresholds of 0.762 and 0.783. This can be explained by the fact that the highest-ranked relevant document is often still retrieved. As the MRR metric focuses primarily on the position of the first relevant context within the retrieved list, the impact of adjusting the threshold is minimized, resulting in similar MRR scores across both conditions.

5.3 Table

Method	**Accuracy(%)**	**Precision**	**Recall**	**F1 score**	**MAP**	**MRR**
LLM-only	71.05	-	-	-	-	-
Naive RAG (0.762)	81.58	0.41	0.70	0.44	1.54	0.83
Naive RAG (0.783)	85.09	0.53	0.58	0.49	1.38	0.78
HyDE RAG (0.762)	79.82	0.30	0.79	0.37	1.53	0.80
HyDE RAG (0.783)	82.46	0.37	0.75	0.41	1.49	0.81

Table 5.1: Comparison of different RAG systems using proposed metrics with 0.762 or 0.783 relevance score threshold using GPT-3.5

Method	Accuracy(%)	Precision	Recall	F1 score	MAP	MRR
LLM-only	85.96	-	-	-	-	-
Naive RAG (0.762)	91.23	0.41	0.70	0.44	1.54	0.83
Naive RAG (0.783)	88.60	0.53	0.58	0.49	1.38	0.78
HyDE RAG (0.762)	93.86	0.33	0.79	0.39	1.60	0.85
HyDE RAG (0.783)	91.23	0.38	0.75	0.43	1.56	0.84

Table 5.2: Comparison of different RAG systems using proposed metrics with 0.762 or 0.783 relevance score threshold using GPT-4o-mini

Table 5.2 and 5.1 compile all the previous data.

Chapter 6

Conclusions and Future work

6.1 Conclusions

In this paper, we propose a method to evaluate RAG systems in medical QA tasks. This work is preliminarily composed of consideration of settings for evaluation, constructing various RAG systems and defining metrics for the evaluation, developing a benchmark dataset, and exploring the effect of different systems, models, and retrieval settings.

The work undertaken in this thesis has important implications. Through implementing and comparing systems such as Naive RAG, HyDE RAG, and LLM-only, we can carry out the extended comparison of their performance in QA tasks. The result demonstrates that RAG systems substantially outperform baseline models, highlighting that combining retrieval techniques with LLM has promise for making the generated answers more reliable. Our development of evaluation metrics tailored to medical RAG systems ensures that the assessment captures the nuances of medical information retrieval, providing a robust approach for future evaluations.

The created benchmark dataset, BioASQ-QA-Y/N, is a critical resource for evaluating how well the RAG systems perform in answering yes/no medical questions. This dataset ensures that evaluations are grounded in realistic and challenging scenarios, mirroring real-life medical information needs. Additionally, our exploration of the relevance score threshold reveals its impact on system performance. The changes in the threshold will affect the trade-off between precision and recall, which may lead to variations in the accuracy and reliability of the RAG systems. We also tested two different models and made a comparison analysis.

The present work fully achieves our goals and shows that combining retrieval and generation improves system performance greatly. The choice of retrieval method and the careful tuning of relevance score thresholds play crucial roles in the outcome.

The positive impact of the thesis includes an approach for more reliable evaluation that might be applied in other medical QA RAG systems. The methodology and findings can also inspire the design and evaluation of RAG systems for other domains.

The presented study also shows the need for further explorations. More advanced retrieval techniques should be tested to optimize performance. The variability in effectiveness across different RAG configurations, especially concerning different thresholds of the relevance score, highlights that there is still potential for further optimization and refinement.

6.2 Limitations

Several limitations have been identified in this study. The evaluation focused primarily on Yes/No questions, which may not represent the richness of the medical QA tasks. Future research should consider expanding the evaluation to include multiple-choice questions and other formats that allow golden standard answers. This will allow for testing the RAG systems for a wide range of question types.

Additionally, due to the substantial workload associated with human annotations, the evaluation only considers accuracy as a metric for generation. This limited scope means that other important aspects, such as understandability, relevance, and the risk assessment of the generated answers, are not thoroughly evaluated. These factors are crucial for ensuring that the responses generated by RAG systems are correct, useful, and safe for patients.

The study also focuses exclusively on medical QA, meaning the findings and methods may not directly apply to other domains. Developing similar benchmarks and evaluation frameworks for different fields will require further research.

Lastly, the project relies solely on OpenAI's GPT model for testing the RAG systems. While these models are state-of-the-art, exploring other LLMs could provide additional insights and potentially reveal different strengths and weaknesses in the RAG systems. Expanding the range of models used in evaluations would help to generalize the findings and improve the robustness of the proposed evaluation framework.

6.3 Ethical Considerations

This paper investigates the evaluation of RAG systems in medical QA tasks, an area that demands a careful and rigorous ethical approach given the sensitivity of medical information. The integration of RAG systems with LLMs for providing medical information guarantees that the produced response is correct and reliable, as any misleading or incorrect information could have profound implications for patient care.

In this work, ethical guidelines have been followed to protect the integrity and dependability of the whole study process. The benchmark dataset used in this study is BioASQ-QA-Y/N, adapted from an official dataset without disclosing sensitive or personally identifiable information. This study also aligns with established data protection regulations.

Moreover, it is important to critically review the potential biases in LLMs and RAG systems so that generated medical answers would not lead to misinformation or health disparities. The evaluation of different settings of RAG systems aims to raise the final goal of a more accurate and reliable way of retrieving medical information by considering the ethical objective of enhancing technology that benefits patients.

6.4 Future work

6.4.1 Expanding the benchmark dataset

To capture the complexity of medical QA tasks more comprehensively, future research should expand the evaluation framework to diversely typed questions beyond Yes/No queries. Examples include multiple-choice questions and other formats that provide golden standard answers. The extension will enable a more thorough evaluation of the capabilities of the RAG systems.

6.4.2 Leveraging Advanced Metrics

Future research should also consider more advanced evaluation metrics to cover a range of dimensions in the performance of the RAG systems for understandability, relevance, and risk assessment of the answers generated. These can ensure that the content produced is accurate, comprehensible, and secure for end-users.

6.4.3 Exploring Other LLMs

This project only used the GPT models of OpenAI for testing the RAG systems. Future research should look into other LLMs, which shall be instructive in other respects and may further reveal different strengths and weaknesses within the RAG systems. Increasing the variety of models tested would have allowed generalization and increased robustness of the proposed evaluation framework.

6.4.4 Generalizability to Other Domains

The study focuses totally on medical QA, but such evaluation frameworks for other domains need to be developed to test the generalizability of the suggested metrics and approaches. Further work should consider applying an adapted version of the evaluation method presented in the paper to other domains, like legal, financial, and technical information retrieval, which helps further validate its robustness and applicability across various contexts.

6.4.5 Advanced RAG Techniques

Apply the methods developed in this work to more advanced and state-of-the-art techniques of RAG techniques. This will include taking advantage of recent progress in retrieval algorithms, prompt engineering, and integration methods that still need to be assessed on RAG systems.

6.5 Reflections

Work undertaken in this thesis shows how RAG systems will revolutionize medical information retrieval by returning appropriate and reliable responses to patient queries.

From an economic perspective, improving the efficiency and accuracy of medical RAG systems can reduce healthcare providers' burden and patient education costs.

Socially, developing robust RAG systems can enhance patient engagement and empowerment, enabling individuals to make more informed decisions about their health. Environmentally, the shift towards digital health solutions can reduce the need for physical consultations and associated travel, contributing to a lower carbon footprint.

Ethically, ensuring that RAG systems are trustworthy and transparent is essential since it reduces the possibility of false information. This project underscores the importance of ongoing research and development in creating systems that are not only technically advanced but also socially and ethically responsible.

References

[1] W. X. Zhao, K. Zhou, J. Li, T. Tang, X. Wang, Y. Hou, Y. Min, B. Zhang, J. Zhang, Z. Dong, Y. Du, C. Yang, Y. Chen, Z. Chen, J. Jiang, R. Ren, Y. Li, X. Tang, Z. Liu, P. Liu, J.-Y. Nie, and J.-R. Wen, "A Survey of Large Language Models," Nov. 2023, arXiv:2303.18223 [cs]. [Online]. Available: http://arxiv.org/abs/2303.18223 [Page 1.]

[2] Z. Xu, S. Jain, and M. Kankanhalli, "Hallucination is Inevitable: An Innate Limitation of Large Language Models," Jan. 2024, arXiv:2401.11817 [cs]. [Online]. Available: http://arxiv.org/abs/2401.11817 [Pages 1 and 12.]

[3] H. Rashkin, V. Nikolaev, M. Lamm, L. Aroyo, M. Collins, D. Das, S. Petrov, G. S. Tomar, I. Turc, and D. Reitter, "Measuring Attribution in Natural Language Generation Models," *Computational Linguistics*, vol. 49, no. 4, pp. 777–840, Dec. 2023. doi: 10.1162/coli_a_00486. [Online]. Available: https://doi.org/10.1162/coli_a_00486 [Page 2.]

[4] S. Es, J. James, L. Espinosa-Anke, and S. Schockaert, "RAGAS: Automated Evaluation of Retrieval Augmented Generation," Sep. 2023, arXiv:2309.15217 [cs]. [Online]. Available: http://arxiv.org/abs/2309.15217 [Pages 2 and 18.]

[5] P. Wang, L. Li, L. Chen, Z. Cai, D. Zhu, B. Lin, Y. Cao, Q. Liu, T. Liu, and Z. Sui, "Large Language Models are not Fair Evaluators," Aug. 2023, arXiv:2305.17926 [cs]. [Online]. Available: http://arxiv.org/abs/2305.17926 [Pages 2 and 18.]

[6] A. Gillioz, J. Casas, E. Mugellini, and O. A. Khaled, "Overview of the Transformer-based Models for NLP Tasks," in *2020 15th Conference on Computer Science and Information Systems (FedCSIS)*, Sep. 2020. doi: 10.15439/2020F20 pp. 179–183. [Online]. Available: https://ieeexplore.ieee.org/abstract/document/9222960 [Page 7.]

[7] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, . u. Kaiser, and I. Polosukhin, “Attention is All you Need,” in *Advances in Neural Information Processing Systems*, vol. 30. Curran Associates, Inc., 2017. [Online]. Available: https://proceedings.neurips.cc/paper/2017/hash/3f5ee243547dee91fbd053c1c4a845aa-Abstract.html [Pages xi, 7, 8, and 9.]

[8] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever *et al.*, “Improving language understanding by generative pre-training,” 2018. [Pages 9 and 10.]

[9] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, “BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding,” May 2019, arXiv:1810.04805 [cs]. [Online]. Available: http://arxiv.org/abs/1810.04805 [Pages 9 and 10.]

[10] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu, “Exploring the Limits of Transfer Learning with a Unified Text-to-Text Transformer,” Sep. 2023, arXiv:1910.10683 [cs, stat]. [Online]. Available: http://arxiv.org/abs/1910.10683 [Page 9.]

[11] Y. Chang, X. Wang, J. Wang, Y. Wu, L. Yang, K. Zhu, H. Chen, X. Yi, C. Wang, Y. Wang, W. Ye, Y. Zhang, Y. Chang, P. S. Yu, Q. Yang, and X. Xie, “A Survey on Evaluation of Large Language Models,” *ACM Transactions on Intelligent Systems and Technology*, vol. 15, no. 3, pp. 39:1–39:45, Mar. 2024. doi: 10.1145/3641289. [Online]. Available: https://dl.acm.org/doi/10.1145/3641289 [Page 9.]

[12] R. Shams, “Semi-supervised Classification for Natural Language Processing,” Sep. 2014, arXiv:1409.7612 [cs] version: 1. [Online]. Available: http://arxiv.org/abs/1409.7612 [Page 10.]

[13] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, L. Kaiser, and I. Polosukhin, “Attention Is All You Need,” Aug. 2023, arXiv:1706.03762 [cs]. [Online]. Available: http://arxiv.org/abs/1706.03762 [Page 10.]

[14] A. Radford, J. Wu, R. Child, D. Luan, D. Amodei, I. Sutskever *et al.*, “Language models are unsupervised multitask learners,” *OpenAI blog*, vol. 1, no. 8, p. 9, 2019. [Page 10.]

[15] 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*, vol. 33. Curran Associates, Inc., 2020, pp. 1877–1901. [Online]. Available: https://proceedings.neurips.cc/paper/2020/hash/1457c0d6bfcb4967418bfb8ac142f64a-Abstract.html [Page 10.]

[16] S. Makridakis, F. Petropoulos, and Y. Kang, "Large Language Models: Their Success and Impact," *Forecasting*, vol. 5, no. 3, pp. 536–549, Sep. 2023. doi: 10.3390/forecast5030030 Number: 3 Publisher: Multidisciplinary Digital Publishing Institute. [Online]. Available: https://www.mdpi.com/2571-9394/5/3/30 [Page 11.]

[17] T. Teubner, C. M. Flath, C. Weinhardt, W. van der Aalst, and O. Hinz, "Welcome to the Era of ChatGPT et al." *Business & Information Systems Engineering*, vol. 65, no. 2, pp. 95–101, Apr. 2023. doi: 10.1007/s12599-023-00795-x. [Online]. Available: https://doi.org/10.1007/s12599-023-00795-x [Page 11.]

[18] OpenAI, J. Achiam, S. Adler, S. Agarwal, L. Ahmad, I. Akkaya, F. L. Aleman, D. Almeida, J. Altenschmidt, S. Altman, S. Anadkat, R. Avila, I. Babuschkin, S. Balaji, V. Balcom, P. Baltescu, H. Bao, M. Bavarian, J. Belgum, I. Bello, J. Berdine, G. Bernadett-Shapiro, C. Berner, L. Bogdonoff, O. Boiko, M. Boyd, A.-L. Brakman, G. Brockman, T. Brooks, M. Brundage, K. Button, T. Cai, R. Campbell, A. Cann, B. Carey, C. Carlson, R. Carmichael, B. Chan, C. Chang, F. Chantzis, D. Chen, S. Chen, R. Chen, J. Chen, M. Chen, B. Chess, C. Cho, C. Chu, H. W. Chung, D. Cummings, J. Currier, Y. Dai, C. Decareaux, T. Degry, N. Deutsch, D. Deville, A. Dhar, D. Dohan, S. Dowling, S. Dunning, A. Ecoffet, A. Eleti, T. Eloundou, D. Farhi, L. Fedus, N. Felix, S. P. Fishman, J. Forte, I. Fulford, L. Gao, E. Georges, C. Gibson, V. Goel, T. Gogineni, G. Goh, R. Gontijo-Lopes, J. Gordon, M. Grafstein, S. Gray, R. Greene, J. Gross, S. S. Gu, Y. Guo, C. Hallacy, J. Han, J. Harris, Y. He, M. Heaton, J. Heidecke, C. Hesse, A. Hickey, W. Hickey, P. Hoeschele, B. Houghton, K. Hsu, S. Hu, X. Hu, J. Huizinga, S. Jain, S. Jain, J. Jang, A. Jiang, R. Jiang, H. Jin, D. Jin, S. Jomoto, B. Jonn,

H. Jun, T. Kaftan, . Kaiser, A. Kamali, I. Kanitscheider, N. S. Keskar, T. Khan, L. Kilpatrick, J. W. Kim, C. Kim, Y. Kim, J. H. Kirchner, J. Kiros, M. Knight, D. Kokotajlo, . Kondraciuk, A. Kondrich, A. Konstantinidis, K. Kosic, G. Krueger, V. Kuo, M. Lampe, I. Lan, T. Lee, J. Leike, J. Leung, D. Levy, C. M. Li, R. Lim, M. Lin, S. Lin, M. Litwin, T. Lopez, R. Lowe, P. Lue, A. Makanju, K. Malfacini, S. Manning, T. Markov, Y. Markovski, B. Martin, K. Mayer, A. Mayne, B. McGrew, S. M. McKinney, C. McLeavey, P. McMillan, J. McNeil, D. Medina, A. Mehta, J. Menick, L. Metz, A. Mishchenko, P. Mishkin, V. Monaco, E. Morikawa, D. Mossing, T. Mu, M. Murati, O. Murk, D. Mély, A. Nair, R. Nakano, R. Nayak, A. Neelakantan, R. Ngo, H. Noh, L. Ouyang, C. O'Keefe, J. Pachocki, A. Paino, J. Palermo, A. Pantuliano, G. Parascandolo, J. Parish, E. Parparita, A. Passos, M. Pavlov, A. Peng, A. Perelman, F. d. A. B. Peres, M. Petrov, H. P. d. O. Pinto, Michael, Pokorny, M. Pokrass, V. H. Pong, T. Powell, A. Power, B. Power, E. Proehl, R. Puri, A. Radford, J. Rae, A. Ramesh, C. Raymond, F. Real, K. Rimbach, C. Ross, B. Rotsted, H. Roussez, N. Ryder, M. Saltarelli, T. Sanders, S. Santurkar, G. Sastry, H. Schmidt, D. Schnurr, J. Schulman, D. Selsam, K. Sheppard, T. Sherbakov, J. Shieh, S. Shoker, P. Shyam, S. Sidor, E. Sigler, M. Simens, J. Sitkin, K. Slama, I. Sohl, B. Sokolowsky, Y. Song, N. Staudacher, F. P. Such, N. Summers, I. Sutskever, J. Tang, N. Tezak, M. B. Thompson, P. Tillet, A. Tootoonchian, E. Tseng, P. Tuggle, N. Turley, J. Tworek, J. F. C. Uribe, A. Vallone, A. Vijayvergiya, C. Voss, C. Wainwright, J. J. Wang, A. Wang, B. Wang, J. Ward, J. Wei, C. J. Weinmann, A. Welihinda, P. Welinder, J. Weng, L. Weng, M. Wiethoff, D. Willner, C. Winter, S. Wolrich, H. Wong, L. Workman, S. Wu, J. Wu, M. Wu, K. Xiao, T. Xu, S. Yoo, K. Yu, Q. Yuan, W. Zaremba, R. Zellers, C. Zhang, M. Zhang, S. Zhao, T. Zheng, J. Zhuang, W. Zhuk, and B. Zoph, "GPT-4 Technical Report," Mar. 2024, arXiv:2303.08774 [cs]. [Online]. Available: http://arxiv.org/abs/2303.08774 [Page 11.]

[19] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W.-t. Yih, T. Rocktäschel, S. Riedel, and D. Kiela, "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," in *Advances in Neural Information Processing Systems*, vol. 33. Curran Associates, Inc., 2020, pp. 9459–9474. [Online]. Available: https://proceedings.neurips.cc/paper/2020/hash/6b493230205f780e1bc26945df7481e5-Abstract.html [Page 12.]

[20] Y. Zhang, H. Fei, and P. Li, "ReadsRE: Retrieval-Augmented Distantly Supervised Relation Extraction," in *Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval*, ser. SIGIR '21. New York, NY, USA: Association for Computing Machinery, Jul. 2021. doi: 10.1145/3404835.3463103. ISBN 978-1-4503-8037-9 pp. 2257–2262. [Online]. Available: https://doi.org/10.1145/3404835.3463103 [Page 12.]

[21] J. Xu, J.-M. Crego, and J. Senellart, "Boosting Neural Machine Translation with Similar Translations," in *Annual Meeting of the Association for Computational Linguistics*. Seattle, United States: Association for Computational Linguistics, Jul. 2020. doi: 10.18653/v1/2020.acl-main.143 pp. 1570–1579. [Online]. Available: https://hal.science/hal-02956324 [Page 12.]

[22] J. Weston, E. Dinan, and A. H. Miller, "Retrieve and Refine: Improved Sequence Generation Models For Dialogue," Sep. 2018, arXiv:1808.04776 [cs]. [Online]. Available: http://arxiv.org/abs/1808.04776 [Page 12.]

[23] 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," Mar. 2024, arXiv:2312.10997 [cs]. [Online]. Available: http://arxiv.org/abs/2312.10997 [Pages xi and 12.]

[24] C. Si, Z. Gan, Z. Yang, S. Wang, J. Wang, J. Boyd-Graber, and L. Wang, "Prompting GPT-3 To Be Reliable," Feb. 2023, arXiv:2210.09150 [cs]. [Online]. Available: http://arxiv.org/abs/2210.09150 [Page 13.]

[25] J. White, Q. Fu, S. Hays, M. Sandborn, C. Olea, H. Gilbert, A. Elnashar, J. Spencer-Smith, and D. C. Schmidt, "A Prompt Pattern Catalog to Enhance Prompt Engineering with ChatGPT," Feb. 2023, arXiv:2302.11382 [cs]. [Online]. Available: http://arxiv.org/abs/2302.11382 [Page 13.]

[26] I. LangChain, "Langchain introduction page," 2024. [Online]. Available: https://python.langchain.com/docs/introduction/ [Page 14.]

[27] "RetrievalTutorials/tutorials/LevelsOfTextSplitting/5_levels_of_text_splitting.ipynb at main · FullStackRetrieval-com/RetrievalTutorials." [Online]. Available: https://github.com/FullStackRetrieval-com/RetrievalTutori

als/blob/main/tutorials/LevelsOfTextSplitting/5_Levels_Of_Text_Splitting.ipynb [Pages xi and 15.]

[28] L. Zheng, W.-L. Chiang, Y. Sheng, S. Zhuang, Z. Wu, Y. Zhuang, Z. Lin, Z. Li, D. Li, E. Xing, H. Zhang, J. E. Gonzalez, and I. Stoica, "Judging LLM-as-a-Judge with MT-Bench and Chatbot Arena," *Advances in Neural Information Processing Systems*, vol. 36, pp. 46 595–46 623, Dec. 2023. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2023/hash/91f18a1287b398d378ef22505bf41832-Abstract-Datasets_and_Benchmarks.html [Page 18.]

[29] Y. Wang, Z. Yu, Z. Zeng, L. Yang, C. Wang, H. Chen, C. Jiang, R. Xie, J. Wang, X. Xie, W. Ye, S. Zhang, and Y. Zhang, "PandaLM: An Automatic Evaluation Benchmark for LLM Instruction Tuning Optimization," May 2024, arXiv:2306.05087 [cs]. [Online]. Available: http://arxiv.org/abs/2306.05087 [Page 18.]

[30] J. Wang, Y. Liang, F. Meng, Z. Sun, H. Shi, Z. Li, J. Xu, J. Qu, and J. Zhou, "Is ChatGPT a Good NLG Evaluator? A Preliminary Study," Oct. 2023, arXiv:2303.04048 [cs]. [Online]. Available: http://arxiv.org/abs/2303.04048 [Page 18.]

[31] C. Zakka, A. Chaurasia, R. Shad, A. R. Dalal, J. L. Kim, M. Moor, K. Alexander, E. Ashley, J. Boyd, K. Boyd, K. Hirsch, C. Langlotz, J. Nelson, and W. Hiesinger, "Almanac: Retrieval-Augmented Language Models for Clinical Medicine," May 2023, arXiv:2303.01229 [cs]. [Online]. Available: http://arxiv.org/abs/2303.01229 [Page 18.]

[32] M. Jeong, J. Sohn, M. Sung, and J. Kang, "Improving medical reasoning through retrieval and self-reflection with retrieval-augmented large language models," *Bioinformatics*, vol. 40, no. Supplement_1, pp. i119–i129, Jul. 2024. doi: 10.1093/bioinformatics/btae238. [Online]. Available: https://doi.org/10.1093/bioinformatics/btae238 [Page 18.]

[33] A. Naik, S. Parasa, S. Feldman, L. L. Wang, and T. Hope, "Literature-Augmented Clinical Outcome Prediction," Nov. 2022, arXiv:2111.08374 [cs]. [Online]. Available: http://arxiv.org/abs/2111.08374 [Pages 19 and 47.]

[34] M. Saeidi, M. Bartolo, P. Lewis, S. Singh, T. Rocktäschel, M. Sheldon, G. Bouchard, and S. Riedel, "Interpretation of Natural Language Rules

in Conversational Machine Reading," Aug. 2018, arXiv:1809.01494 [cs, stat]. [Online]. Available: http://arxiv.org/abs/1809.01494 [Page 27.]

[35] G. Tsatsaronis, G. Balikas, P. Malakasiotis, I. Partalas, M. Zschunke, M. R. Alvers, D. Weissenborn, A. Krithara, S. Petridis, D. Polychronopoulos, Y. Almirantis, J. Pavlopoulos, N. Baskiotis, P. Gallinari, T. Artiéres, A.-C. N. Ngomo, N. Heino, E. Gaussier, L. Barrio-Alvers, M. Schroeder, I. Androutsopoulos, and G. Paliouras, "An overview of the BIOASQ large-scale biomedical semantic indexing and question answering competition," *BMC Bioinformatics*, vol. 16, no. 1, p. 138, Apr. 2015. doi: 10.1186/s12859-015-0564-6. [Online]. Available: https://doi.org/10.1186/s12859-015-0564-6 [Page 31.]

[36] L. Gao, X. Ma, J. Lin, and J. Callan, "Precise Zero-Shot Dense Retrieval without Relevance Labels," Dec. 2022, arXiv:2212.10496 [cs]. [Online]. Available: http://arxiv.org/abs/2212.10496 [Page 39.]

TRITA – EECS-EX 2024:0000
Stockholm, Sweden 2024

www.kth.se

€€€€ For DIVA €€€€

{
"Author1": { "Last name": "Han",
"First name": "Tangyujun",
"Local User Id": "u100001",
"E-mail": "tanghan@kth.se",
"organisation": {"L1": "School of Electrical Engineering and Computer Science",
}
},
"Cycle": "2",
"Course code": "DA246X",
"Credits": "30.0",
"Degree1": {"Educational program": "Master's Programme, Communication Systems, 120 credits"
,"programcode": "TCOMM"
,"Degree": "Master's Programme, Communication Systems, 120 credits"
,"subjectArea": "Computer Science and Engineering"
},
"Title": {
"Main title": "Evaluation of Retrieval-Augmented Generation in Medical Question Answering Tasks",
"Language": "eng" },
"Alternative title": {
"Main title": "Utvärdering av hämtningsförstärkt generation i medicinska fråge-svar uppgifter",
"Language": "swe"
},
"Supervisor1": { "Last name": "Kheirabadi",
"First name": "Amirhossein Layegh",
"Local User Id": "u100003",
"E-mail": "amlk@kth.se",
"organisation": {"L1": "School of Electrical Engineering and Computer Science",
"L2": "DIVISION OF SOFTWARE AND COMPUTER SYSTEMS" }
},
"Examiner1": { "Last name": "Payberah",
"First name": "Amir H.",
"Local User Id": "u1d13i2c",
"E-mail": "payberah@kth.se",
"organisation": {"L1": "School of Electrical Engineering and Computer Science",
"L2": "DIVISION OF SOFTWARE AND COMPUTER SYSTEMS" }
},
"National Subject Categories": "10201, 10206",
"Other information": {"Year": "2024", "Number of pages": "1,65"},
"Copyrightleft": "copyright",
"Series": { "Title of series": "TRITA – EECS-EX" , "No. in series": "2024:0000" },
"Opponents": { "Name": "A. B. Normal & A. X. E. Normalè"},
"Presentation": { "Date": "2022-03-15 13:00"
,"Language":"eng"
,"Room": "via Zoom https://kth-se.zoom.us/j/ddddddddddd"
,"Address": "Isafjordsgatan 22 (Kistagången 16)"
,"City": "Stockholm" },
"Number of lang instances": "2",
"Abstract[eng]": €€€€

```
% Write an abstract that is about 250 and 350 words (1/2 A4-page)  with the following components:
% key parts of the abstract
% \begin{itemize}
%   \item What is the topic area? (optional) Introduces the subject area for the project.
%   \item Short problem statement
%   \item Why was this problem worth a Bachelor's/'Masters thesis project? (\ie, why is the problem
both significant and of a suitable degree of difficulty for a Bachelor's/'Masters thesis project? Why
has no one else solved it yet?)
%   \item How did you solve the problem? What was your method/insight?
%   \item Results/Conclusions/Consequences/Impact: What are your key
results/\linebreak[4]conclusions? What will others do based on your results? What can be done now
that you have finished - that could not be done before your thesis project was completed?
% \end{itemize}
```

Recent developments and changes in Large Language Models (LLMs) have great potential for application in the field of medical question answering (QA), particularly through Retrieval-Augmented Generation (RAG) systems. These systems address challenges in providing reliable and personalized medical information by integrating authoritative sources. However, evaluating their performance remains a critical challenge, especially in sensitive medical contexts where accuracy is critical. Current evaluation techniques often rely on heavy human annotations, making the process time-consuming and labor-intensive. While using LLMs as evaluators has been proposed as an alternative to reduce the manual workload, its reliability remains questionable.

This thesis introduces a new evaluation method to solve this problem, tested by constructing various RAG systems, including Naive RAG and Hypothetical Document Embeddings (HyDE) RAG. The evaluation leverages two different LLMs and is based on a benchmark dataset specifically designed for yes/no medical questions, with an LLM-only system serving as the baseline. Metrics used for evaluation include Accuracy, Precision, Recall, F1 score, Mean Accuracy (MAP), and Mean Reciprocity Rating (MRR) to measure retrieval and generation performance comprehensively. In addition, the study explored the impact of different search relevance thresholds and different models on the RAG system, providing insights for further optimization.

The experimental results show that RAG systems greatly improve the accuracy and reliability of medical information retrieval compared to baseline models. The choice of retrieval relevance thresholds and the selection of different LLMs also impact the performance of RAG systems. The paper proposes a robust evaluation method for RAG systems in medical QA and lays the foundation for extending this method into other knowledge-intensive domains. Such reliable evaluations will contribute to developing more effective and reliable medical QA systems, benefiting both healthcare providers and patients.

€€€€,
”Keywords[eng]”: €€€€
Medical Question Answering, Large Language Models, Retrieval-Augmented Generation, Evaluation €€€€,
”Abstract[swe]”: €€€€

% \engExpl{If you are writing your thesis in English, you can leave this until the draft version that goes to your opponent for the written opposition. In this way, you can provide the English and Swedish abstract/summary information that can be used in the announcement for your oral presentation.\\If you are writing your thesis in English, then this section can be a summary targeted at a more general reader. However, if you are writing your thesis in Swedish, then the reverse is true – your abstract should be for your target audience, while an English summary can be written targeted at a more general audience.\\This means that the English abstract and Swedish sammnfattning % or Swedish abstract and English summary need not be literal translations of each other.}

% \warningExpl{Do not use the \textbackslash glspl\{\} command in an abstract that is not in English, as my programs do not know how to generate plurals in other languages. Instead, you will need to spell these terms out or give the proper plural form. In fact, it is a good idea not to use the glossary commands at all in an abstract/summary in a language other than the language used in the \texttt{acronyms.tex file} – since the glossary package does \textbf{not} support use of more than one language.}

Den senaste tidens utveckling och förändringar inom Stora språkmodeller (LLMs) har stor potential för tillämpning inom området medicinsk frågesvar (QA), särskilt genom Retrieval-Augmented Generation (RAG) system. Dessa system hanterar utmaningar när det gäller att tillhandahålla tillförlitlig och personlig medicinsk information genom att integrera auktoritativa källor. Att utvärdera deras prestanda är dock fortfarande en stor utmaning, särskilt i känsliga medicinska sammanhang där noggrannhet är avgörande. Nuvarande utvärderingstekniker förlitar sig ofta på tunga mänskliga kommentarer, vilket gör processen tidskrävande och arbetsintensiv. Att använda LLM:er som utvärderare har föreslagits som ett alternativ för att minska den manuella arbetsbelastningen, men dess tillförlitlighet är fortfarande tveksam.

Denna avhandling introducerar en ny utvärderingsmetod för att lösa detta problem, testad genom att konstruera olika RAG-system, inklusive Naive RAG och Hypothetical Document Embeddings (HyDE) RAG. Utvärderingen baseras på en referensdatauppsättning som är särskilt utformad för medicinska ja/nej-frågor, med ett LLM-only-system som fungerar som baslinje. Mätvärden som används för utvärdering inkluderar noggrannhet, precision, återkallande, F1 poäng, genomsnittlig noggrannhet (MAP) och genomsnittlig ömsesidighet (MRR) för att på ett heltäckande sätt mäta prestanda för hämtning och generering. Dessutom undersökte studien effekterna av olika tröskelvärden för sökrelevans och olika modeller på RAG-systemet, vilket gav insikter för ytterligare optimering.

De experimentella resultaten visar att RAG-systemen kraftigt förbättrar noggrannheten och tillförlitligheten vid medicinsk informationssökning jämfört med baslinjemodeller. Valet av tröskelvärden för hämtningsrelevans och valet av olika LLM påverkar också RAG-systemens prestanda. I artikeln föreslås en robust utvärderingsmetod för RAG-system inom medicinsk kvalitetssäkring och grunden läggs för att utvidga denna metod till andra kunskapsintensiva domäner. Sådana tillförlitliga utvärderingar kommer att bidra till utvecklingen av mer effektiva och tillförlitliga medicinska kvalitetssäkringssystem, vilket gynnar både vårdgivare och patienter.

€€€€,
”Keywords[swe]”: €€€€
medicinsk frågesvar, Stora språkmodeller, Retrieval-Augmented Generation, Utvärdering. €€€€,
}

acronyms.tex

```
%%% Local Variables:
%%% mode: latex
%%% TeX-master: t
%%% End:
% The following command is used with glossaries-extra
\setabbreviationstyle[acronym]{long-short}
% The form of the entries in this file is \newacronym{label}{acronym}{phrase}
%                                       or \newacronym[options]{label}{acronym}{phrase}
% see ”User Manual for glossaries.sty” for the  details about the options, one example is shown below
% note the specification of the long form plural in the line below
% \newacronym[longplural={Debugging Information Entities}]{DIE}{DIE}{Debugging Information Entity}
% %
% % The following example also uses options
% \newacronym[shortplural={OSes}, firstplural={operating systems (OSes)}]{OS}{OS}{operating system}

% % note the use of a non-breaking dash in long text for the following acronym
% \newacronym{IQL}{IQL}{Independent -QLearning}

% % example of putting in a trademark on first expansion
% \newacronym[first={NVIDIA OpenSHMEM Library (NVSHMEM\texttrademark)}]{NVSHMEM}{NVSHMEM}{NVIDIA
     OpenSHMEM Library}

\newacronym{KTH}{KTH}{KTH Royal Institute of Technology}

\newacronym{LAN}{LAN}{Local Area Network}
\newacronym{VM}{VM}{virtual machine}
% note the use of a non-breaking dash in the following acronym
\newacronym{WiFi}{-WiFi}{Wireless Fidelity}

\newacronym{WLAN}{WLAN}{Wireless Local Area Network}
\newacronym{UN}{UN}{United Nations}
\newacronym{SDG}{SDG}{Sustainable Development Goal}
\newacronym{AI}{AI}{Artificial Intelligence}
\newacronym{API}{API}{Application Interface}
\newacronym{GPT}{GPT}{Generative Pre-trained Transformer}
\newacronym{HyDE}{HyDE}{Hypothetical Document Embeddings}
\newacronym{LLM}{LLM}{Large Language Model}
\newacronym{LSTM}{LSTM}{Long Short-Term Memory}
\newacronym{MAP}{MAP}{Mean Average Precision}
\newacronym{ML}{ML}{Machine Learning}
\newacronym{MRR}{MRR}{Mean Reciprocal Rank}
\newacronym{NLG}{NLG}{Natural Language Generation}
\newacronym{NLP}{NLP}{Natural Language Processing}
\newacronym{QA}{QA}{Question Answering}
\newacronym{RAG}{RAG}{Retreival-Augmented Generation}
\newacronym{RNN}{RNN}{Recurrent Neural Networks}
```