SENTIMENTCAREBOT: CHATBOT À GÉNÉRATION AUGMENTÉE PAR RECHERCHE POUR LE SOUTIEN EN SANTÉ MENTALE AVEC INTÉGRATION DES SENTIMENTS

Mémoire présenté

dans le cadre du programme de maîtrise en informatique

en vue de l'obtention du grade de maître ès sciences (M.Sc.)

PAR

©JEAN PIERRE NAYINZIRA

AVRIL 2025

RÉSUMÉ

Le système mondial de soins de santé mentale fait face à divers défis en matière d'accessibilité et de disponibilité du soutien spécialisé, tels que les psychologues et les conseillers, notamment à la suite de la pandémie de COVID-19. Cette étude explore une solution potentielle à ce problème en développant un modèle de chatbot, *SentimentCare-Bot*, qui intègre l'analyse des sentiments avec des techniques de la génération augmentée de récupération (RAG) et des modèles de langage avancés (LLMs). L'étude utilise un ensemble de données publiques de «Mental Health Counseling Conversations »et des méthodes de sélection de bases telles que «Naive RAG », «Multi-query RAG »et «Hypothetical Document Embeddings »(HyDE) pour améliorer les traductions de requêtes. Les résultats du test de «Tukey's Honest Significant Difference »(HSD) révèlent une amélioration significative des performances de l'analyse des sentiments lorsqu'elle est appliquée au «Multi-query RAG »utilisant le modèle de langage MistralAI, comparé au «Multi-query RAG »utilisant le modèle de langage d'OpenAI et à HyDE utilisant OpenAI avec l'analyse des sentiments. Ces résultats démontrent le potentiel de l'analyse des sentiments pour améliorer l'efficacité des chatbots de santé mentale.

ABSTRACT

The global mental healthcare system faces various challenges in terms of accessibility and the availability of specialist support, such as psychologists and counselors, especially following the COVID-19 pandemic. This thesis explores a potential solution to this problem by developing a chatbot model, *SentimentCareBot*, which integrates sentiment analysis with Retrieved-Augmented Generation (RAG) techniques and large language models (LLMs). The study uses a public available Mental Health Counseling Conversations Dataset and baseline selection methods such as Naive RAG, Multi-query RAG, and Hypothetical Document Embeddings (HyDE) to improve query translations. The findings from Tukey's Honest Significant Difference (HSD) test reveals a significant improvement in sentiment analysis performance when it is applied to the Multi-query RAG using the MistralAI language model, compared to both Multi-query RAG using the OpenAI language model and HyDE using OpenAI with Sentiment Analysis. These results demonstrate the potential of sentiment analysis to enhance the effectiveness of mental health chatbots.

TABLE OF CONTENTS

LIST OF TABLES

LIST OF FIGURES

LIST OF ABBREVIATIONS

AI Artificial Intelligence

AIML Artificial Intelligence Markup Language

FAISS Facebook AI Similarity Search

GPT Generative Pre-trained Transformer

HyDE Hypothetical Document Embedding

LLM Large Language Model

LSTM Long Short-Term Memory

NLP Natural Language Processing

NLU Natural Language Understanding

RAG Retrieval-Augmented Generation

RNNs Recurrent Neural Networks

Seq2Seq Sequence-to-Sequence

WHO World Health Organization

DEDICATION

To my family, whose unwavering support and encouragement have been my source of strength throughout this journey.

ACKNOWLEDGEMENTS

I am deeply grateful to everyone who has supported and guided me throughout this journey. First and foremost, I would like to thank my thesis advisor, Mehdi Adda, whose expertise, patience, and insightful feedback have been invaluable. Their mentorship has challenged me to think critically and refine my ideas, helping me shape this work into what it is today.

To my family and friends, thank you for your endless encouragement, understanding, and patience, especially during the more challenging times. Your unwavering support has been my foundation. I am truly blessed to have such a supportive network, without whom this journey would not have been possible.

PREFACE

This thesis represents the culmination of my graduate studies in Information Technology at Université du Québec à Rimouski. My research was driven by a desire to explore the integration of sentiment analysis with Chatbots in mental healthcare applications. This field has rapidly evolved in recent years, opening new possibilities and challenges that inspired me to investigate and contribute to its growth.

The following chapters present the research process, methodology, findings, and insights gathered during my study. This work contributes to ongoing discussions and developments in Chatbots, encouraging further research and innovation. I am excited to share this journey with the academic community and I am eager to see how this work may inspire future research and applications.

GENERAL INTRODUCTION

0.1 CONTEXT

The World Health Organization (WHO) brings attention to the critical importance of mental health as a fundamental human well-being, by promoting the improvement of global mental health. They focus on the significant gap between the demand for mental health services and their current availability, despite different public health crises, in particular the COVID-19 pandemic. WHO recommends integrating mental health services into primary healthcare and achieving universal health coverage (Organization *et al.*, 2022). In addition, the WHO points out the essential role of healthcare professionals in expanding mental healthcare, recommending increased training and support networks to improve service provision. They call on governments to boost mental health funding to facilitate these necessary changes.

0.2 PROBLEMATIC

Despite the recognized potential and demonstrated effectiveness of chatbots in mental healthcare, several significant challenges persist. A key issue is the ethical concern regarding the reliability and effectiveness of these chatbots. Many existing chatbots lack robust evidence-based support or sufficient research backing to confirm their effectiveness in various mental health contexts. This gap raises concerns about the ethical deployment of chatbots for mental health purposes. Additionally, chatbots inherently lack the ability to experience and convey empathy as humans do, which is a fundamental component of effective mental health support (Denecke *et al.*, 2021). While some users perceive chatbots like Woebot as empathetic and supportive, this perception is not universal, leading to varying user experiences (Boucher *et al.*, 2021). The challenge lies in addressing these limitations to enhance the reliability, effectiveness, and empathetic capabilities of mental health chatbots.

0.3 OBJECTIVES AND RESEARCH QUESTIONS

This research aims to enhance the effectiveness of mental health chatbots by integrating sentiment analysis into Retrieval-Augmented Generation (RAG) models, leading to the development of *SentimentCareBot*. The chatbot is designed to detect emotional signals and generate contextually appropriate responses based on users' emotional states.

0.3.1 SUB-OBJECTIVES

1. **Sentiment Classification**: Improve the chatbot's ability to accurately recognize and classify user sentiments to ensure emotionally appropriate responses.

2. **Evaluation of RAG Models**: Assess the performance of different RAG models when integrated with sentiment analysis to determine their impact on chatbot effectiveness.

3. **Comparative Analysis**: Analyze the variations in faithfulness, answer relevancy, and correctness across different RAG models.

4. **Statistical Validation**: Employ ANOVA and Tukey's HSD test to validate the differences in sentiment analysis performance across various RAG models.

0.4 METHODOLOGY

This study begins with a comprehensive review of the literature on natural language processing (NLP) and natural language understanding (NLU). The review examined the current state-of-the-art of conversational agents, commonly known as chatbots, to justify our model selection. Additionally, we explored Retrieval-Augmented Generation (RAG) models (Lewis *et al.*, 2021; Jiang *et al.*, 2023; Li *et al.*, 2022; Gao *et al.*, 2024) and their variations to establish the foundation for our approach.

To contextualize our choice of the RAG model, we analyzed advancements in NLP and NLU, particularly their applications in the healthcare domain and their influence on chatbot evolution. This historical analysis traced the progression of chatbot development, highlighting key proposals that addressed previous limitations and introduced novel methodologies. Special emphasis was placed on the role of the attention mechanism, which has significantly enhanced chatbot sophistication and human-like conversational abilities.

A crucial aspect of this study involved understanding the transformer model (Vaswani *et al.*, 2017), which employs multiple attention mechanisms to improve performance. The development of large language models (LLMs) (Zhao *et al.*, 2023) was discussed, focusing on their ability to manage long-distance dependencies in text through parallel computation. Despite their capabilities, LLMs are prone to hallucinations, a limitation that RAG mitigates by updating knowledge bases and ensuring factual accuracy.

We presented a detailed analysis of RAG components to elucidate their functionalities and establish a foundation for integrating sentiment analysis into mental health chatbots. Subsequently, we outlined the baseline selection of the RAG models used in this study. The role of reranking in refining chatbot responses was also examined, mainly in relation to sentiment analysis.

For this study, we utilized the "Mental Health Counseling Conversations Dataset" available on Hugging Face (Bertagnolli, 2020). This dataset comprises 3,512 question-answer pairs sourced from two online counseling and therapy platforms. The evaluation dataset included 106 question-answer pairs, representing approximately 3% of the entire dataset, while the remaining 97% was used to build our knowledge base.

To assess the performance of the *SentimentCareBot* architecture (Figure 0.1), we implemented a baseline selection through query translation and used a vector database to simplify

retrieval through similarity search. Then a sentiment analysis ranker was applied to filter the retrieved documents based on their relevance and sentiment score. The re-ranked documents, along with the initial input query, were provided to the LLM to generate the final response. We evaluated its performance using the Retrieval Augmented Generation Assessment (Ragas) (Es *et al.*, 2023) metrics, specifically *Faithfulness*, *Answer Relevancy*, and *Answer Correctness* scores. Two LLMs were selected for evaluation: OpenAI's "gpt-3.5-turbo-0125" and MistralAI's "mistral-large-latest." Sentiment analysis was integrated into each scenario to measure its impact on performance. In addition, an API performance analysis was conducted, focusing on token usage and latency using a subset of five question-answer pairs.

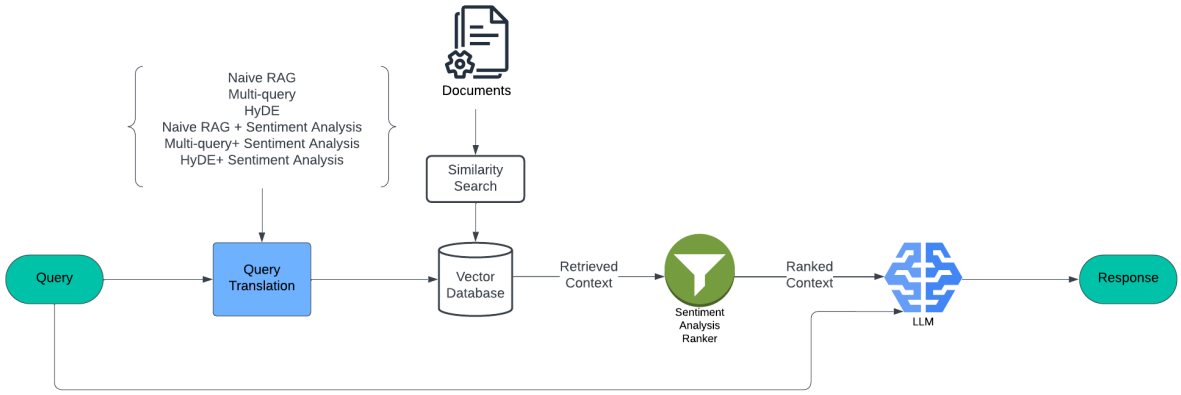

Figure 0.1 : SentimentCareBot Architecture

Finally, an ANOVA test was conducted to determine whether significant differences exist in *Answer Relevancy* across different RAG models. Tukey's Honestly Significant Difference (Tukey's HSD) test was then applied to assess statistical variations between configurations. This analysis was aiming to evaluate the influence of sentiment analysis and language model selection on the overall performance of RAG models.

0.5 CONTRIBUTIONS

This research explores Retrieval-Augmented Generation (RAG) models for mental healthcare by integrating sentiment analysis to improve response relevance in mental health conversations. It develops a sentiment-aware RAG framework that refines query expansion, retrieval, and response generation. A comparative analysis of Naïve RAG, Multi-query RAG, and Hypothetical Document Embedding (HyDE) evaluates their effectiveness, while a sentiment-sensitive retrieval reranking mechanism further optimizes response quality. Furthermore, this study explores mental health conversation datasets and conducts empirical evaluations, demonstrating improved accuracy, sentiment alignment, and chatbot responsiveness, contributing to emotionally aware AI systems for mental health support.

Unlike conventional RAG models that focus on lexical relevance, this research introduces sentiment-driven retrieval and reranking strategies to ensure responses align with users' emotional contexts. This approach improves the responsiveness of chatbots in mental health conversations, where contextual understanding and emotional sensitivity are critical.

The findings of this research contribute to the advancement of empathetic and context-aware AI-driven mental health applications. Notably, the SentimentCareBot (Nayinzira & Adda, 2024) research paper was published in the proceeding of the ICTH-24 conference, highlighting its significance.

0.6 ORGANISATION

This thesis by articles is structured into three chapters, framed by a general introduction and a concluding section. The general introduction sets the stage by presenting the overall context of the subject, while the conclusion reflects on the opportunities emerging from our research.

1. **Article 1: Comprehensive Literature Review on Retrieval-Augmented Generation (RAG) Chatbots for Mental Health Support**: The first chapter is a research paper that provides a comprehensive overview of the current state-of-the-art of natural language processing (NLP) and natural language understanding (NLU), their intersections with the healthcare domain, the evolution of chatbots, and Retrieval-Augmented Generation (RAG). This paper aims to explain the various concepts employed in our study to reinforce the relevance of our research topic. Article status: finalized.

2. **Article 2: SentimentCareBot: Retrieval-Augmented Generation Chatbot for Mental Health Support with Sentiment Analysis**: The second chapter features the research paper accepted for presentation at the ICTH-24 conference (Nayinzira & Adda, 2024), including its abstract in French. The paper is presented in the same format that the original was submitted and published. Article status: published.

3. **Results and Discussion**: This chapter is dedicated to presenting a deeper analysis on the obtained results. This includes an examination of baseline selections, the significant difference between models and the token usage and latency metrics of each model.

CHAPTER I

ARTICLE 1: COMPREHENSIVE LITERATURE REVIEW ON RETRIEVAL-AUGMENTED GENERATION (RAG) CHATBOTS FOR MENTAL HEALTH SUPPORT

1.1 RÉSUMÉ EN FRANÇAIS DE L'ARTICLE

Cette étude explore les avancées cruciales du traitement de la langue naturel (NLP), de la compréhension de la langue naturel (NLU) et de la génération augmentée de récupération (RAG) dans le domaine de la santé, ainsi que l'évolution de l'intelligence artificielle (IA) conversationnelle. Elle met en avant le rôle du NLP et du NLU dans le traitement des données non structurées, améliorant ainsi les soins aux patients et l'efficacité des services de santé. L'évolution des agents conversationnels, des modèles basés sur des règles comme Eliza aux systèmes d'IA tels qu'Alexa, est discutée parallèlement aux avancées en IA. L'étude analyse également la capacité de la RAG à améliorer la précision des agents en intégrant la récupération de connaissances avec des modèles génératifs. Elle également aborde les défis tels que la désinformation et la confidentialité des données, ainsi que son impact potentiel sur le secteur de la santé.

1.2 COMPREHENSIVE LITERATURE REVIEW ON RETRIEVAL-AUGMENTED GENERATION (RAG) CHATBOTS FOR MENTAL HEALTH SUPPORT

Comprehensive Literature Review on Retrieval-Augmented Generation (RAG) Chatbots for Mental Health Support

Jean Pierre Nayinzira
Département de mathématiques, informatique et génie
University of Quebec at Rimouski
Programme de Maîtrise en Informatique
Email : JeanPierre.Nayinzira@uqar.ca

Mehdi Adda
Département de mathématiques, informatique et génie
University of Quebec at Rimouski
Programme de Maîtrise en Informatique
Email : mehdi.adda@gmail.com

***Résumé*—This review explores the critical advancements in natural language processing (NLP), natural language understanding (NLU), and Retrieval-Augmented Generation (RAG) in healthcare and the evolution of conversational Artificial Intelligence (AI). It highlights NLP and NLU's role in processing unstructured data, enhancing patient care and healthcare efficiency. The evolution of chatbots, from rule-based models like Eliza to AI-driven systems like Alexa, is discussed alongside machine learning advancements. The study also analyzes RAG's ability to improve AI accuracy by integrating knowledge retrieval with generative models. It addresses challenges such as misinformation and data privacy and its potential impact on healthcare.**

1. Introduction

The relationship between natural language processing (NLP) and natural language understanding (NLU) plays a crucial role in the transformation of the healthcare sector by enabling more sophisticated data processing and improving patient-system interactions. This study explores the evolution of chatbot technologies within this domain, tracing advancements from early models such as Eliza and PARRY to modern AI-driven platforms like Amazon's Alexa. A particular focus is placed on the development of Retrieval-Augmented Generation (RAG) frameworks, which integrate internal knowledge with external information retrieval to enhance the accuracy and contextual relevance of generated responses.

Recent advancements in large language models (LLMs) and retrieval-augmented generation (RAG) have significantly enhanced natural language understanding and generation, leading to their widespread application in various domains, including healthcare. Existing reviews, such as those by Naveed et al. [21] and Minaee et al. [20], provide comprehensive overviews of LLM architectures, augmentation techniques, and chatbot functionalities. However, while these studies acknowledge the role of LLM-powered chatbots in healthcare for tasks like patient query responses and appointment scheduling, they do not extensively explore the use of sentiment analysis in mental health applications. Additionally, discussions on retrieval mechanisms often focus on general improvements in accuracy and efficiency rather than their specific implications for mental healthcare contexts, where ethical considerations and patient safety are paramount.

Despite the growing interest in using LLMs and RAG for healthcare, there remains a significant gap in research concerning their application in mental health support, particularly in integrating sentiment analysis with RAG-based chatbots. Gao et al. [8] provide a detailed discussion on various RAG paradigms and their role in enhancing language models but do not address their potential in mental health settings. Furthermore, while existing reviews discuss chatbot development in general, they lack focused analyses on how retrieval-based enhancements can improve sentiment detection and response generation for mental health conversations. Given the sensitivity of mental health interactions, an in-depth exploration of sentiment-aware RAG implementations is necessary to ensure ethical, effective, and contextually appropriate chatbot responses. This review aims to fill this critical gap by synthesizing relevant research on the intersection of RAG, sentiment analysis, and mental healthcare.

In this review, we explore the state-of-the-art definitions of natural language processing (NLP) and natural language understanding (NLU), highlighting their importance and applications in the healthcare domain. Furthermore, we explore the evolution of chatbots to understand their fundamental role in the development of large language models (LLMs). In addition, we discuss the fundamental building blocks of LLMs, including word embeddings and transformers, which have been pivotal in their advancement. Finally, we introduce Retrieval-Augmented Generation (RAG) as a solution to the limitations of LLMs. In our exploration of RAG, we analyze its key components, various query transformation baselines, and the role of reranking in improving retrieval effectiveness.

2. Definition of Natural Language Processing and Natural Language Understanding

2.1. Natural Language Processing

Natural Language Processing (Figure 1) is a branch of machine learning that enables the processing and analysis of free text. It handles text or speech input as complex syntactic and phonological data to extract meaning and generate quantitative outputs. NLP incorporates techniques like Natural Language Understanding (NLU) and Natural Language Generation (NLG) for applications such as machine translation, question-answering, and chatbots [17].

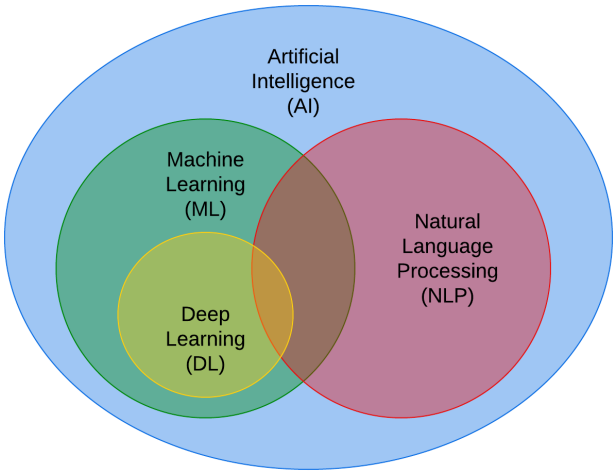

FIGURE 1 – Natural Language Processing

NLP encompasses various techniques to process and understand natural language, with NLU being a subset of NLP that focuses on understanding human language [19]. NLP involves the development of algorithms and models that enable computers to process and understand human language in a way that is both meaningful and useful. The primary goal of NLP is to enable machines to read, decipher, understand, and make sense of human languages in a valuable manner [16].

Additionally, NLP is a group of methods and computer-aided algorithms designed to detect patterns in textual data. By treating words and clusters of words as meaningful, by extracting concepts and relationships from texts more efficiently than humans are capable of doing. It is considered a valuable strategy for conducting content analyses of academic literature, through methods such as bibliometric and scientometric studies. These methods involve examining digital data objects to quantify study characteristics within a publication dataset [25].

2.2. Natural Language Understanding

Natural Language Understanding (Figure 2) is a primary NLP technique used for text categorization, information extraction tasks, and semantic analysis. It involves analyzing linguistic features including phonology, morphology, syntax, and semantics. NLU is essential for applications like text categorization, information extraction, and semantic analysis [17].

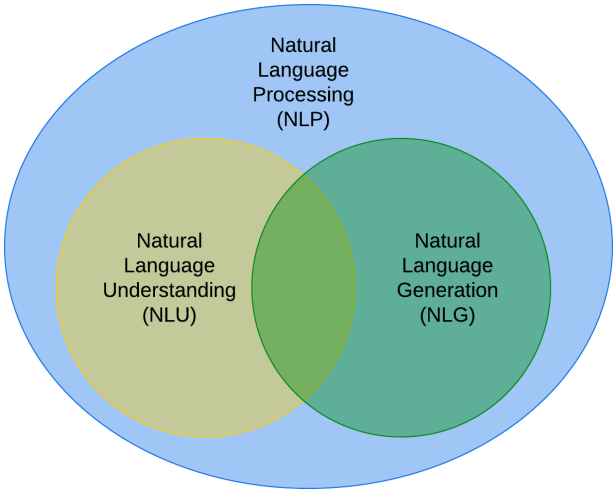

FIGURE 2 – Natural Language Understanding

NLU is a subset of NLP and conversational AI that helps computers understand human language by understanding, analyzing, and interpreting basic message or speech parts. NLU is trained with natural user utterances tagged with entities and expanded with the use of synonyms. NLU services use high-level APIs to build language parsers using existing NLU and machine learning libraries. The main task of NLU is intent and entity detection to understand the user input and the intent of the conversation [19].

NLU focuses on machine reading comprehension, going beyond merely processing text to understanding the meaning and context of the words. It involves interpreting the input text, recognizing the intent behind the text, and extracting relevant information [16].

Furthermore, NLU is mentioned in the context of its relationship with NLP, highlighting that there is still a significant gap between the two. While NLP focuses on detecting patterns and extracting information from text, NLU aims to capture the nuance in language usage and how people process information. Cutting-edge research such as OpenAI's GPT-3 is moving closer to NLU benchmarks by using large amounts of text-based data to better capture language nuances. However, NLU has not yet advanced to the point where it can fully replace human understanding [25].

3. The Importance of Natural Language Processing and Natural Language Understanding in Healthcare Domain

Natural Language Processing (NLP) and Natural Language Understanding (NLU) are pivotal in transforming

healthcare through their ability to process, analyze, and generate human language. These technologies facilitate the extraction of meaningful information from vast amounts of unstructured data, enhance patient care, and improve healthcare system efficiency. This section discusses various significant studies of NLP and NLU in healthcare.

Enhancing Patient Care and Interaction - NLP and NLU are involved in developing patient-facing applications such as chatbots, which provide efficient and user-friendly interfaces for patient interaction. These applications can answer questions, conduct initial consultations, and automate the response process, thus streamlining healthcare processes. Examples include mobile applications like Babylon Health and HealthTap, and specialized chatbots like 'Pharmabot' and 'Mandy,' which explain medications and facilitate patient interviews [17], [19].

Unlocking Unstructured Data - A significant portion of Electronic Health Records (EHRs) consists of unstructured data such as clinical notes and discharge summaries. NLP processes and converts these free-text elements into structured data, enabling clinicians to evaluate treatments and interventions more effectively. This transformation is crucial for making sense of vast amounts of patient information and improving clinical outcomes [17], [16].

Improving Predictive Models in Critical Care - NLP enhances predictive models in critical care by extracting detailed information from free-text notes, which improves predictions of patient outcomes. This application is vital for identifying patients suitable for critical care trials and improving recruitment efficiency for clinical studies. Additionally, NLP aids in generating comprehensive problem lists from EHRs, significantly boosting patient safety and reducing delays and costs [17], [16].

Augmenting Clinical Research and Coding - NLP supports clinical research by facilitating the search for relevant clinical trials and streamlining drug discovery processes. It also aids in clinical coding by processing unstructured data, which helps in evaluating the efficacy of treatments and interventions. This capability is particularly vital in the context of large databases and systematic reviews [17], [16].

Efficiency in Healthcare Systems - The automation of various healthcare processes, such as response systems and patient interviews, significantly improves overall efficiency. For example, NLP-based systems can predict hospital admissions from the Emergency Department, augmenting the response process and improving clinical outcomes [17], [19]. NLP's ability to synthesize large volumes of literature also aids in identifying trends and topics across numerous articles, supporting decision-making and quality improvement initiatives [25].

Analyzing Linguistic Features for Better Understanding - NLP's strength lies in its ability to analyze linguistic features, including phonology, morphology, syntax, and semantics. This analysis is crucial for applications such as text categorization, information extraction, and semantic analysis, which are essential for improving medical research and patient care. Advanced language models, such as OpenAI's GPT-3, are making strides in capturing these nuances, which is crucial for accurately interpreting and applying healthcare research [25].

Supporting Public Health Initiatives - NLP and NLU have been instrumental in supporting public health initiatives, especially during health crises such as the COVID-19 pandemic. Chatbots like "COVID-19 Info Serbia" have been developed to provide citizens with reliable and up-to-date information, reducing the burden on healthcare helplines and ensuring timely and accurate information dissemination. These technologies have enhanced tasks such as text classification, information retrieval, and knowledge discovery, aiding in the dissemination and understanding of critical health information [19], [16].

In summary, NLP and NLU play crucial roles in healthcare by enabling the processing of unstructured data, enhancing patient care, and improving system efficiency. They also help to understand different healthcare sectors by supporting decision-making and highlighting their importance. As these technologies evolve, they will drive significant growth and improve health outcomes.

4. Chatbot evolution

With the proposal of the Turing test by Alan Turing in 1950, which posed the question "Can machines think ?", the foundation for chatbot development was laid, signaling the start of their popularization. The first notable chatbot, Eliza, developed in 1966, mimicked a psychotherapist by employing simple pattern matching and template-based responses to echo user input as questions. Despite its rudimentary conversational abilities, Eliza managed to confuse people during an era when computer interaction was rare, thus encouraging the creation of more sophisticated chatbots. Significant advancements was achieved with the introduction of PARRY in 1972, a chatbot designed with a distinct personality. Later, the 1995 creation of the chatbot ALICE, which was awarded the "most human computer" title after winning the Loebner Prize in 2000, 2001, and 2004, showcased further progress. ALICE used a pattern matching algorithm enhanced by the Artificial Intelligence Markup Language (AIML), facilitating the expansion of its knowledge base by developers [1].

Machine learning and natural language processing advancements have facilitated the development of advanced chatbots, including Amazon's Alexa and Google's Assistant, reducing reliance on rules and pattern matching. These innovations have improved chatbot flexibility, ease of implementation, and the precision of human-like conversation, presenting clear advantages over rule-based models in terms of adaptability and domain independence. Furthermore, machine learning enables chatbots to learn directly from human dialogue, eliminating the need to manually define patterns, and thus increasing their adaptability [4].

Within the framework of Seq2Seq (Sequence-to-Sequence) models, the attention mechanism plays a crucial role in overcoming the limitations posed by encoding entire input sentences into fixed-length context vectors. This innovative mechanism enables the decoder to selectively

focus on various parts of the input sequence, effectively preserving vital information and context that could be lost in longer sequences. By forming direct connections between the target and source, the attention mechanism allows the model to concentrate on relevant segments of the input during translation or response generation, thereby enhancing the model's capability to handle variable-length sequences and improving the simplicity and clarity of generated responses. Such advancements make the attention mechanism a crucial element in the development of more sophisticated and human-like conversational agents.

Further insights on the evolution of chatbot technologies are transformer models that employ multiple attention mechanisms and demonstrate better performance and accuracy over traditional Seq2Seq models equipped with attention mechanisms. The key to the transformer model's success lies in its ability to address sequence-related challenges without relying on recurrent neural networks (RNNs), thus optimizing training time and enhancing neural machine translation performance. Nuanced handling and interpretation of input sequences facilitated by the attention mechanisms within transformers result in more accurate and contextually relevant responses. This leap forward marks a significant step in the ongoing effort to harness advanced neural network techniques, including Long-Short-Term Memory (LSTM) cells [18], [26].

The Transformer Architecture, which employs a self-attention mechanism to process complex data sequences, is central to the development of LLMs. This design facilitates parallel computations and effectively manages long-distance dependencies in text, laying the groundwork for the Generative Pre-trained Transformer (GPT) series. Notably, these models excel in generating text that mirrors human quality and achieve high accuracy in various linguistic tasks, thanks to extensive training on broad textual corpora [5].

Despite these technological strides, integrating LLMs into healthcare presents distinct challenges, particularly medical misinformation. Concerns include the misinterpretation of medical terms and the generation of advice that contradicts standard medical practices. Furthermore, the use of commercial servers by LLMs to store patient health data has sparked debates about privacy implications [31].

Addressing the limitations of LLMs, the RAG framework offers a distinct approach by tackling the challenges of updating knowledge bases, clarifying prediction rationales, and correcting factual inaccuracies. By combining a pre-trained sequence-to-sequence model with a detailed context index, RAG ensures more accurate and fact-based context generation. This innovative method not only reduces the likelihood of misinformation, but also enhances the interpretability and adaptability of the model, setting a new standard for knowledge-intensive natural language processing tasks [14], [12], [15], [8].

5. Language Models

Language models have evolved from simple statistical tools into advanced techniques capable of understanding and generating human-like language. The scaling of LLMs has enabled models to acquire capabilities such as in-context learning, instruction following, and complex reasoning, which were previously unattainable. These advancements have significant implications for various applications, from improving search engines to developing AI chatbots that engage in meaningful dialogues with users [31].

The development and deployment of LLMs also pose challenges, such as the need for vast computational resources and the potential for generating harmful content. However, with ongoing research and engineering efforts, LLMs continue to push the boundaries of what is possible in artificial intelligence.

5.1. Word Embedding

Word embedding is a method for representing words in a continuous vector space, mapping semantically similar words to nearby points. This method captures semantic relationships between words, such as synonyms and analogies. The technique significantly influences natural language processing (NLP), especially in tasks like text classification, information retrieval, and translation [24].

5.2. Transformer

Transformers have revolutionized the field of NLP by introducing a model architecture that relies entirely on self-attention mechanisms, bypassing the need for recurrent or convolutional layers traditionally used in sequence modeling tasks. [27] first introduced this breakthrough model in their seminal work "Attention is All You Need" in 2017, where they proposed the Transformer architecture designed to handle dependencies between input and output sequences without regard to their distance. The fundamental innovation in the Transformer model is its ability to perform parallel computations, significantly reducing training times while improving performance across various NLP tasks.

The Transformer model (Figure 4) is built upon an encoder-decoder structure, where both components are composed of stacked layers of self-attention and fully connected feed-forward networks. The encoder processes the input sequence and generates a continuous representation, which the decoder then uses to produce the output sequence. Each layer in the encoder and decoder has two sub-layers : a multi-head self-attention mechanism and a position-wise fully connected feed-forward network. The self-attention mechanism is critical to the Transformer's ability to capture contextual relationships between words in a sequence, regardless of their position, making it particularly powerful for tasks like machine translation.

A key feature of the Transformer model is its multi-head attention mechanism, which allows the model to focus on different parts of the input sequence simultaneously. By projecting the input into multiple subspaces, the model can learn various aspects of the input data, enhancing its ability to capture complex dependencies. Additionally, since the Transformer does not inherently encode the order

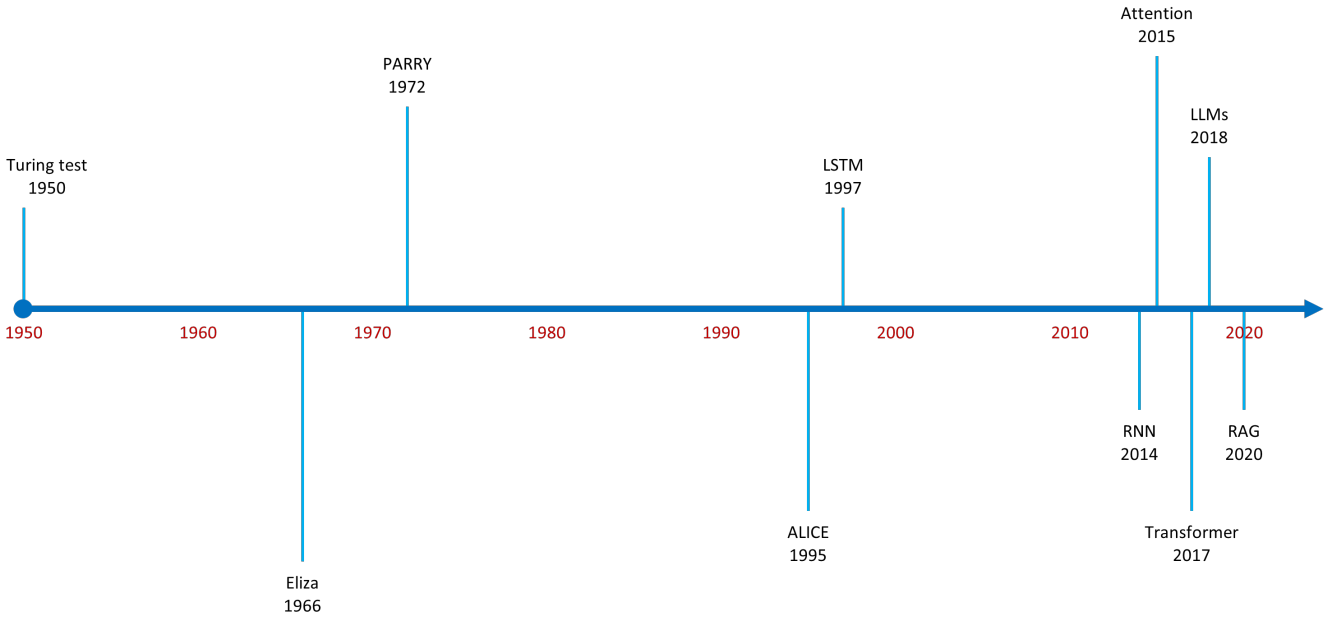

FIGURE 3 – Chatbot evolution timeline.

of sequences, positional encodings are added to the input embeddings to provide the model with information about the relative position of each token in the sequence.

Building on the success of the original Transformer architecture, Devlin et al. [6] introduced BERT (Bidirectional Encoder Representations from Transformers) in 2019, which extends the capabilities of Transformers by pre-training deep bidirectional representations from unlabeled text. Unlike earlier models that only processed sequences in a unidirectional manner, BERT captures context from both directions, making it particularly effective for tasks requiring an understanding of the relationship between different parts of a text, such as question answering and language inference.

5.3. Pre-trained Language Models

Pre-trained language models (PTMs) have revolutionized the field of NLP by shifting the paradigm from traditional supervised learning methods to a more efficient and scalable approach that involves pre-training followed by fine-tuning. These models leverage large amounts of unannotated data to learn general language representations through self-supervised learning techniques, which can then be adapted to various downstream tasks such as text classification, named entity recognition, and machine translation [28].

5.4. Large Language Models

From the transformer models, this led to the development of Large Language Models (LLMs), such as GPT-3, PaLM, and LLaMA, which represented the next leap in this evolution. These models are characterized by their enormous scale, often containing tens or hundreds of billions of parameters. This scaling has proven to improve performance on a wide range of tasks, including those previously unsolvable by smaller models.

Despite their capabilities, the training and deployment of LLMs pose significant challenges. Training requires extensive computational resources and sophisticated techniques to ensure stability and effectiveness. Moreover, aligning LLMs with human values is crucial to prevent generating harmful or biased content [31].

6. Retrieval-Augmented Generation

Retrieval-Augmented Generation (RAG) is an advanced method designed to enhance the factual correctness and relevance of generated content by integrating external knowledge from retrieved documents into the generation process. This approach is crucial for addressing the inherent limitations of large language models (LLMs) such as hallucinations and outdated knowledge, thereby grounding the generated output in up-to-date factual information from external sources [2].

The RAG framework employs retrieval mechanisms to obtain a list of relevant documents that provide contextual knowledge to support the generation process. This integration is intended to reduce the likelihood of hallucinations and update the information with the latest research [2], [13], [7]. The typical RAG workflow includes three essential steps : corpus partitioning and vector indexing, identifying and retrieving chunks based on vector similarity, and generating a response conditioned on the retrieved chunks [8].

RAG combines both the innate knowledge of LLMs and additional information fetched from external documents

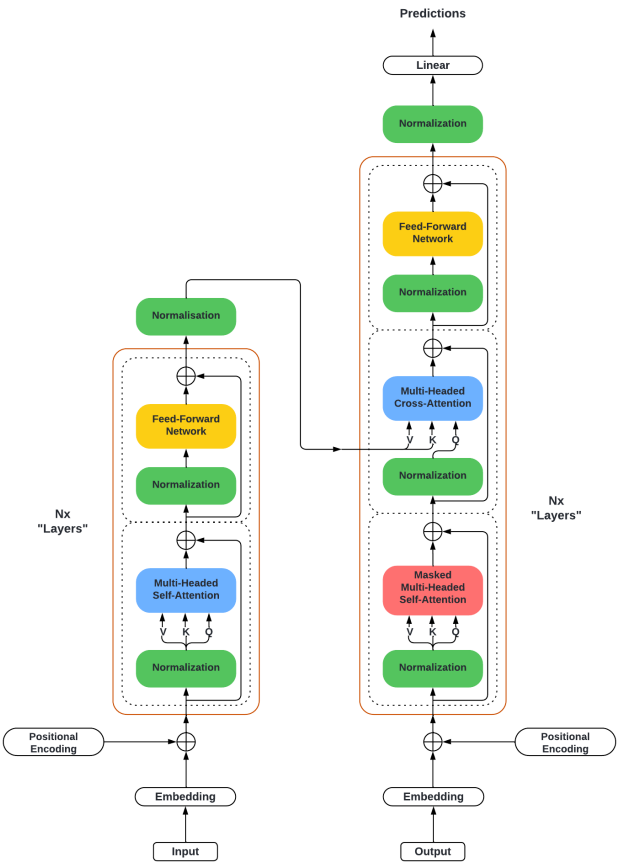

FIGURE 4 – The transformer architecture inspired by Vaswani et al [27].

to generate more accurate and contextually rich responses. This dual reliance on extensive background documents and generative capabilities ensures comprehensive responses to user queries, particularly in knowledge-dense NLP tasks. As a result, RAG often outperforms conventional sequence-to-sequence (seq2seq) models and certain retrieve-and-extract architectures [13].

Technological advancements in RAG have led to the development of innovative approaches addressing critical questions such as what to retrieve, when to retrieve, and how to use the retrieved information. These techniques aim to optimize the retrieval process, improve the integration of retrieved information into the generative model, and ensure more relevant and precise outputs [8].

The evolution of RAG has progressed through various phases, from Naive RAG to Advanced RAG and Modular RAG. Each phase introduced enhancements to address limitations in retrieval, generation, and augmentation techniques. Conceptually, RAG is defined as a paradigm that enhances LLMs by integrating external knowledge bases, combining information retrieval mechanisms and in-context learning to bolster the LLM's performance without the need for retraining for task-specific applications [8].

RAG represents a significant advancement in the field of NLP by effectively bridging the gap between static LLMs and dynamic external knowledge sources. By leveraging both parametric and non-parametric memory, such as pre-trained seq2seq transformers and dense vector indexes, RAG enables models to generate informed, contextually relevant, and accurate responses across various applications [14].

6.1. RAG Components

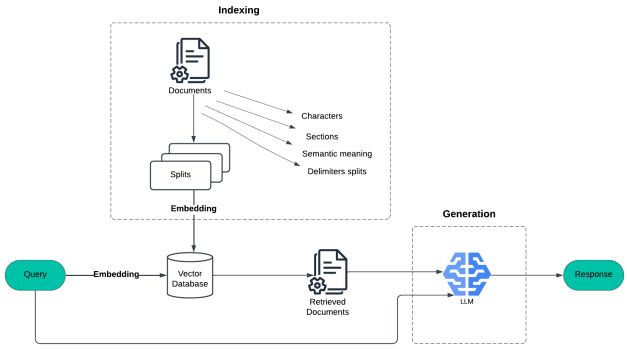

FIGURE 5 – RAG Components.

Figure 5 illustrates the components of Retrieval-Augmented Generation (RAG), comprising three main steps : indexing, retrieval, and generation.

6.1.1. Indexing

Retrieval Augmented Generation indexing is an advanced methodology that integrates various processes to enhance the retrieval and generation capabilities of LLMs. From Agarwal et al.'s method [2], the RAG indexing process begins with the use of an LLM to summarize user-provided abstracts into keywords. These keywords are then employed to query a search engine, which retrieves relevant papers. The retrieved documents are reranked by an LLM-based reranker based on their relevance to the abstract, and the reranked papers are used as context to generate the related work section of a paper.

Kim & Min [13] elaborate on the document preprocessing phase of RAG indexing. Initially, documents are processed using Optical Character Recognition (OCR) technology to convert them into text. This text is divided into chunks to facilitate better indexing and retrieval. Following this, the documents are embedded using models like the LLM-Embedder, transforming text chunks into vector representations for similarity searches. The similarity search is conducted using Facebook AI Similarity Search (FAISS), known for its efficiency and scalability, particularly in large-scale datasets, enhancing retrieval accuracy and relevance in specialized domains such as pharmaceutical regulatory compliance.

Eibich et al. [7] describe the RAG indexing process through the "Document Summary Index" method, which improves RAG systems by indexing document summaries

for efficient retrieval while providing full-text documents to LLMs for response generation. This decoupling strategy optimizes retrieval speed and accuracy through summary-based indexing and supports comprehensive response synthesis by utilizing the original text.

Lewis et al. [14] discusses the use of a dense vector index of Wikipedia built using FAISS with a Hierarchical Navigable Small World (HNSW) approximation for fast retrieval in RAG indexing. Each Wikipedia article is split into disjoint 100-word chunks, creating a total of 21 million documents. The document encoder computes an embedding for each document, which is then indexed for maximum inner product search (MIPS). This method enables efficient retrieval of the top K documents relevant to a query, which are then used as additional context for generating the target sequence.

In essence, RAG indexing involves keyword generation, document retrieval, reranking, and embedding, leveraging technologies like OCR, FAISS, and various LLMs to optimize both retrieval accuracy and the generation of relevant and comprehensive responses.

6.1.2. Retrieval

RAG retrieval is an advanced methodology designed to enhance the relevance and precision of information retrieval, primarily to augment the performance of generative models. This approach integrates various creative techniques to ensure that the retrieved information is highly relevant, accurate, and contextually extensive.

According to Kim & Min [13] method, the core of RAG retrieval involves a dual-track retrieval process that leverages both the user's original query and a hypothetical answer generated by a fine-tuned large language model (LLM). This dual approach broadens the search scope by capturing a wider array of potentially relevant information. The system retrieves half of the documents using the original query and the other half using the hypothetical answer, ensuring a more thorough and nuanced retrieval of information. This method is particularly useful in specialized domains like pharmaceutical regulatory compliance.

A critical component of RAG retrieval is the use of similarity search, particularly through tools like Facebook AI Similarity Search (FAISS). FAISS provides efficient and scalable similarity search capabilities, which are advantageous for handling large-scale datasets and offer significant improvements in retrieval speed and accuracy.

The reranking process plays a vital role in enhancing the relevance of retrieved documents. After the initial retrieval, documents are reranked based on their relevance scores with respect to the query. Only those with the highest relevance scores are retained, ensuring that the most pertinent information is prioritized.

Eibich et al. [7] use of Hypothetical Document Embedding (HyDE) is another key technique used in RAG retrieval. HyDE involves generating a hypothetical answer to a query using LLMs and embedding this answer to refine and focus document retrieval efforts. This enhances the ability to produce context-rich answers and improves retrieval precision.

RAG retrieval also incorporates methods like sentence-window retrieval and multi-query retrieval to enhance its capabilities. Sentence-window retrieval focuses on using small data chunks, such as single sentences, to improve retrieval performance. Multi-query retrieval expands a single user query into multiple similar queries, each subjected to its own retrieval process. This increases the chances of fetching a higher volume of relevant information.

Maximal Marginal Relevance (MMR) further refines the retrieval process by balancing relevance and diversity in the documents retrieved. It evaluates documents for their closeness to the query's intent and their uniqueness compared to already selected documents, reducing redundancy and covering a broader range of information.

Gao et al. [8] method of enhancing semantic representations and fine-tuning embedding models are crucial for optimizing the RAG retrieval process. Breaking down external documents into smaller chunks to extract fine-grained features and embedding these chunks accurately represent their semantics. Fine-tuning embedding models enhances retrieval relevance in domain-specific contexts by generating training data using language models like GPT-3.5-turbo to create question-chunk pairs.

Recursive retrieval refines search queries based on previous search results, enhancing search depth and relevance. This method involves a structured index to process and retrieve data hierarchically, summarizing sections before performing secondary retrievals for more refined searches. Adaptive retrieval dynamically adjusts the retrieval process to meet the specific demands of varying tasks and contexts.

Hybrid search exploration integrates keyword-based, semantic, and vector searches, ensuring consistent retrieval of highly relevant and context-rich information.

Finally, the Dense Passage Retriever (DPR) forms the foundation of RAG retrieval by using a bi-encoder architecture where a dense representation of a document is produced by a BERT-based document encoder, and a query representation is produced by a BERT-based query encoder. The retrieval process is treated as a Maximum Inner Product Search (MIPS) problem, allowing efficient retrieval of the top-K documents most relevant to the input query. This mechanism ensures fast and effective retrieval, using the most pertinent documents as additional context for generating the output [14].

Overall, RAG retrieval's comprehensive and nuanced approach significantly enhances the quality and efficiency of information retrieval, ensuring that the information used for generation is both relevant and contextually appropriate.

6.1.3. Generation

Retrieval-Augmented Generation represents a sophisticated blend of retrieval mechanisms and generative models aimed at creating contextually rich and accurate outputs. This method leverages both retrieved documents and advanced prompting techniques to significantly enhance the quality of generated responses.

At the core of RAG is the integration of retrieved data into the generative model's input. Unlike traditional language models, which rely solely on internal data, RAG incorporates external information obtained through retrieval processes. This comprehensive input, which includes contextual information and relevant text segments from the retriever, allows for a deeper understanding of the query's context. This leads to more informative and contextually relevant responses, as highlighted by Gao et al [8].

One notable technique employed in RAG is few-shot prompting. This approach involves using example question-answer sets provided before the actual question to improve response accuracy. Few-shot prompting has proven to be more effective than zero-shot inference, as it provides the model with a better framework for understanding the query and generating accurate responses [13].

The final answer generation often involves sophisticated models such as ChatGPT-3.5-turbo. These models use the retrieved context and refined prompts to generate the final answer, ensuring that the responses are contextually rich and highly relevant. This stage highlights the importance of using advanced language models that can leverage the nuanced information provided by the retrieval process [13].

A crucial aspect of RAG is the evaluation and reranking of retrieved documents. The documents, obtained through both the original query and hypothetical answers generated by a fine-tuned large language model (LLM), are assessed for relevance. Only the most pertinent documents are used in the final response generation, ensuring that the generated outputs are highly relevant to the user's query [13].

Post-retrieval processing is another essential component of RAG. This stage involves filtering and optimizing the relevant information retrieved to enhance the quality of the results. Techniques such as information compression and result reranking are employed to manage context length restrictions and reduce noise, thus improving the overall coherence and relevance of the generated content [8].

The Hypothetical Document Embedding (HyDE) technique further refines the RAG process. By leveraging large language models to generate hypothetical answers to queries, which are then embedded to focus document retrieval efforts, HyDE enhances the generation phase with context-rich answers. This technique underscores the importance of using sophisticated methods to improve both retrieval and generation stages [7].

Advanced techniques and models play a significant role in the effectiveness of RAG. For instance, RAG employs pre-trained seq2seq models like BART-large as the generator. These models condition the output sequence on the input sequence, retrieved documents, and previously generated tokens. The use of retrieved documents as non-parametric memory provides additional context, resulting in more factual, specific, and diverse generated texts [14].

Guided generation is another critical aspect, where the generator is directed by the retrieved text to ensure coherence between the generated content and the obtained information. This guidance helps refine the model's adaptation to the input data derived from queries and documents, leading to more accurate and relevant responses [8].

In general, RAG represents a powerful approach that integrates advanced retrieval and generative techniques to produce responses that are both accurate and contextually appropriate. By leveraging sophisticated models and methods, RAG enhances the overall performance and relevance of generated outputs, making it a valuable tool in various applications.

6.2. Different Types of RAG

Table 1 illustrates various methodologies designed to improve the Standard RAG, commonly referred to as the "Naive RAG." The Naive RAG framework represents the foundational stage in the evolution of RAG, developed to address inherent limitations of language models, mainly in handling knowledge-intensive tasks. These methodologies primarily focus on improving the retrieval accuracy of a large language model (LLM). By using these techniques as a foundational basis, insights can be drawn on how sentiment analysis can be effectively integrated into mental health chatbots.

6.2.1. Corrective RAG

Corrective Retrieval-Augmented Generation (CRAG) [30] enhances traditional RAG frameworks by integrating a corrective mechanism that evaluates and refines retrieved documents before they influence the generation process. This ensures that only relevant and reliable information shapes chatbot responses, which is especially crucial for sentiment-aware applications in mental health, where accuracy and emotional nuance are essential.

At the core of CRAG is a retrieval evaluator that assesses each document's relevance and trustworthiness. Based on confidence scores, one of three corrective actions is applied : (1) *Correct*, where high-quality retrievals undergo a decompose-then-recompose algorithm for refinement ; (2) *Incorrect*, where unreliable documents are discarded, prompting a large-scale web search for better sources ; and (3) *Ambiguous*, where uncertain cases trigger a hybrid approach, combining internal and external retrievals for robustness. This process improves the accuracy of facts and contextual interpretation, making CRAG particularly suitable for sentiment-aware mental health chatbots.

CRAG's ability to filter unreliable information minimizes the risk of generating misleading responses while allowing dynamic integration of external knowledge. This adaptability ensures that chatbots remain informed and contextually relevant, mainly in mental health conversations, where responses must be emotionally attuned and factually sound.

However, implementing CRAG presents challenges. Its reliance on an external evaluator necessitates domain-specific fine-tuning, especially for detecting factual inaccuracies and sentiment misinterpretations. In addition, the computational overhead of document evaluation, refinement,

RAG Type	Description
Corrective RAG	Uses a lightweight evaluator to assess and adjust retrieval quality. If the retrieval lacks accuracy, CRAG uses web searches [30].
RAG-Fusion	This method incorporates a Reciprocal Rank Fusion (RRF), which generates and reranks multiple queries to provide responses that are more accurate [23].
Self-RAG	This framework allows models to retrieve relevant information on demand and self-reflect on both retrieved and generated content using "reflection tokens [3]."
Graph RAG	GraphRAG uses a structured graph-based relationships between entities to enhance retrieval and generation in RAG, using elements like nodes and paths [22].
Modular RAG	This method decomposes complex RAG systems into independent modules and operators, creating a reconfigurable, LEGO-like framework [9].
Adaptive RAG	This dynamically adjusts retrieval strategies in RAG models by assessing query complexity and selecting between non-retrieval, single-step, and multi-step approaches [11].
Speculative RAG	Speculative RAG leverages a smaller, specialized language model to draft multiple answer candidates from distinct document subsets, while a larger generalist language model verifies these drafts in parallel [29].

TABLE 1 – Different Types of RAG

and external searches can affect response latency, posing challenges for real-time chatbot interactions.

Another concern is the dependency on external web searches, which, while improving retrieval accuracy, also introduces risks related to source reliability and bias. In mental healthcare, where trust is paramount, improper validation of external information could undermine the credibility of chatbots. Careful curation of sources is essential to prevent the introduction of misleading content.

In summary, CRAG significantly advances RAG methodologies by enhancing retrieval accuracy through structured correction mechanisms. Its adaptability makes it ideal for sentiment-aware applications in sensitive domains such as mental health. However, optimizing evaluator training, improving computational efficiency, and ensuring rigorous source validation are critical to maximize the effectiveness of CRAG in conversational AI.

6.2.2. RAG-Fusion

RAG-Fusion [23] extends traditional RAG by incorporating Reciprocal Rank Fusion (RRF) to enhance response accuracy and retrieval efficiency. It employs a multi-query mechanism, expanding an initial user query into semantically related subqueries using a large language model (LLM). This approach captures diverse aspects of the original query, broadening document retrieval coverage.

A key feature of RAG-Fusion is its integration of RRF, a reranking algorithm that refines document selection by scoring relevance across all subqueries. The ranking function applies a reciprocal rank mechanism, accumulating weighted scores to produce a fused list that prioritizes the most relevant sources. This balances query diversity while mitigating biases in individual queries, improving retrieval robustness.

RAG-Fusion is particularly effective for complex inquiries where a single retrieval pass may overlook contextual variations. By iteratively refining the selection of documents, it improves the relevance of the response. However, its reliance on multiple retrieval and reranking stages can introduce latency, potentially impacting real-time chatbot applications. Additionally, excessive divergence in subqueries may lead to the retrieval of tangentially related documents, affecting response coherence.

In sentiment-aware chatbots for mental healthcare, RAG-Fusion offers both opportunities and challenges. Its ability to retrieve contextually diverse information supports nuanced sentiment analysis, which is crucial for conversational agents. Future adaptations incorporating sentiment-aware reranking could further refine emotionally relevant responses. However, increased computational complexity may require optimization strategies, such as adaptive query expansion or reinforcement learning-based reranking, to balance efficiency and accuracy in real-time use.

In general, RAG-Fusion improves traditional RAG by integrating a fusion-based ranking mechanism. Its modular approach to multi-query expansion and document fusion holds promise for sentiment-aware chatbot architectures, provided efficiency trade-offs are effectively managed.

6.2.3. Self-RAG

RAG enhances the factual accuracy and contextual grounding of LLMs by incorporating external knowledge. Traditional RAG methods append retrieved documents to the model's input but rely on fixed retrieval mechanisms that may introduce irrelevant information or fail to adapt dynamically to query requirements. To address these limitations, Self-Reflective Retrieval-Augmented Generation (Self-RAG) [3] introduces an adaptive retrieval process that iteratively refines responses through self-assessment.

Self-RAG distinguishes itself by enabling dynamic retrieval on demand rather than relying on a predefined set of documents. A key feature is the integration of "reflection tokens," which prompt the model to retrieve additional information or assess its response based on relevance, evidential

support, and coherence. This self-reflective process reduces hallucinations and improves factual consistency by refining outputs iteratively.

The Self-RAG framework consists of three core components : (1) a retrieval module that dynamically queries external knowledge, (2) a generation module that synthesizes responses using retrieved information and prior context, and (3) a reflection module that evaluates and refines outputs. This iterative assessment improves accuracy but remains dependent on the model's internal evaluation capabilities, which may still allow some unsupported claims to pass verification.

Integrating Self-RAG into sentiment-aware chatbot applications offers notable benefits, particularly in mental healthcare, where both contextual understanding and factual reliability are crucial. By leveraging iterative retrieval and reflection, such chatbots can generate responses that are contextually appropriate and well-supported, reducing misinformation risks and fostering trust. However, further refinement is needed to enhance consistency and reliability, especially in domains demanding high factual precision.

In general, Self-RAG advances RAG methodologies by incorporating self-reflection to improve accuracy and contextual relevance. While it mitigates limitations of traditional RAG through dynamic retrieval and iterative refinement, ongoing research is necessary to strengthen its ability to detect and correct inconsistencies. Its application in sentiment-aware chatbots highlights its potential for mental healthcare, where precise and emotionally intelligent responses are essential for effective user engagement.

6.2.4. Graph RAG

RAG models enhance large language models by integrating external knowledge sources for more informed response generation. Graph Retrieval-Augmented Generation (GraphRAG) [22] extends this approach by incorporating structured relational knowledge from graph databases. Unlike traditional RAG models that rely on vector-based similarity retrieval, GraphRAG utilizes graph structures to capture semantic dependencies, representing knowledge as nodes, paths, and subgraphs. This enables more context-aware and nuanced responses.

GraphRAG operates through three main stages : (1) *Graph-Based Retrieval*, where entity-linking identifies relevant nodes and subgraphs for a given query ; (2) *Contextualization*, where retrieved graph elements are processed for coherence and interpretability ; and (3) *Generation*, where the LLM synthesizes responses using the retrieved knowledge. This structured approach makes GraphRAG particularly effective in domains requiring structured reasoning, such as healthcare, finance, and law.

Despite its strengths, GraphRAG faces challenges related to efficiency and scalability. Identifying the most relevant subgraph becomes computationally complex as graph size increases, often requiring heuristic search algorithms that may not always yield optimal results. Additionally, GraphRAG depends on continuously updated knowledge graphs, as outdated information can lead to inaccurate responses, particularly in fast-evolving domains.

In sentiment analysis-based chatbots, GraphRAG offers both advantages and constraints. Its structured retrieval mechanism can enhance sentiment-driven responses by incorporating emotional and contextual dependencies from conversation graphs. Sentiment-labeled graphs allow the chatbot to retrieve sentiment-aware knowledge, refining responses to align with user emotions. However, maintaining real-time adaptability requires frequent updates to sentiment graphs, adding computational complexity. Thus, while GraphRAG holds promise for sentiment-aware response generation, its integration demands careful management of knowledge graph curation, efficiency, and adaptability.

6.2.5. Modular RAG

The Modular Retrieval-Augmented Generation (RAG) framework, proposed by Gao et al. [9], advances RAG architectures by replacing traditional monolithic pipelines with highly adaptable, reconfigurable structures. Unlike conventional retrieve-then-generate models, it decomposes RAG into independent functional modules, akin to a LEGO-like system. This modularity enhances flexibility, allowing designers to dynamically configure, extend, and optimize components based on specific needs.

The framework consists of three key layers : retrieval, generation, and coordination. The retrieval module fetches relevant knowledge, while the generation module refines and contextualizes it into coherent outputs. The coordination layer manages integration and workflow, enabling flexible orchestration rather than a fixed retrieval-generation sequence. This adaptability supports parallel and adaptive processing strategies, facilitating modifications and enhancements without disrupting the entire system.

Modular RAG is particularly suited for complex applications, such as sentiment-aware chatbots in mental healthcare. By allowing dynamic reconfiguration, it can integrate sentiment analysis as an additional processing module, ensuring responses are contextually relevant and sentiment-sensitive. Its structure also enables iterative improvements, such as fine-tuned retrieval strategies for emotionally charged discussions.

However, increased modularity introduces challenges in system maintenance, debugging, and workflow optimization due to the complexity of managing interactions between modules. Dependencies must be efficiently handled to minimize computational overhead. These challenges, though, can be addressed through intelligent orchestration and optimization techniques, reinforcing Modular RAG's potential for sentiment-driven chatbot applications.

By dynamically adjusting retrieval and generation based on user sentiment, Modular RAG enhances response coherence and emotional alignment. Despite its engineering complexities, its ability to improve sentiment-adaptive chatbot systems in mental healthcare makes it a promising avenue for further research and development.

6.2.6. Adaptive RAG

RAG enhances chatbot responses by integrating external knowledge retrieval. However, traditional RAG relies on a static retrieval strategy that fails to adapt to query complexity, leading to inefficiencies where simple queries undergo unnecessary retrieval, while complex ones may lack sufficient contextual grounding. To address this, Adaptive Retrieval-Augmented Generation (Adaptive-RAG) [11] dynamically optimizes retrieval strategies based on query complexity, improving efficiency and response relevance.

Adaptive-RAG employs a classifier that categorizes queries into three retrieval strategies : non-retrieval, single-step retrieval, or multi-step retrieval. Simple queries bypass retrieval, relying solely on generation, while moderately complex ones trigger a single-step retrieval for additional information. Highly complex queries undergo multi-step retrieval, iteratively refining retrieved documents to enhance response accuracy. This approach ensures efficient resource allocation while maintaining high-quality responses.

Despite its advantages, Adaptive-RAG depends on a query complexity classifier trained on automatically generated labels, as no standard dataset exists for this task. While this method supports scalability, it introduces potential inaccuracies due to dataset biases and misclassification. If the classifier misjudges query complexity, it may apply a suboptimal retrieval strategy, leading to either insufficient context or unnecessary computational overhead.

In sentiment-aware chatbots, Adaptive-RAG offers a promising enhancement by adjusting retrieval depth based on emotional tone and intent. Sentiment analysis can refine query classification where emotionally charged queries may require multi-step retrieval for well-contextualized, supportive responses, while neutral inquiries might benefit from minimal or no retrieval. Integrating sentiment analysis with Adaptive-RAG can improve contextual awareness, empathy, and efficiency in mental health-related conversations.

While Adaptive-RAG addresses static RAG inefficiencies, its reliance on automatically generated complexity labels remains a challenge. Incorporating sentiment analysis could further refine retrieval strategies, particularly in sentiment-sensitive applications like mental healthcare. Future research should explore hybrid approaches with manually annotated datasets to enhance classifier accuracy, improving Adaptive-RAG's effectiveness in conversational AI.

6.2.7. Speculative RAG

The Speculative Retrieval-Augmented Generation (Speculative RAG) [29] framework enhances accuracy and efficiency in retrieval and response generation. Unlike traditional RAG models that rely on a single language model, Speculative RAG employs a smaller, specialized "RAG drafter" to generate multiple answer drafts. These drafts are based on distinct subsets of retrieved documents, ensuring diverse perspectives and mitigating redundancy and positional bias in extended contexts.

Speculative RAG operates through three key components : (1) the retrieval module, (2) the speculative drafting module, and (3) the final synthesis module. The retrieval module sources relevant documents, which are clustered by content similarity. The drafting module then selects samples from these clusters, optimizing token efficiency and ensuring diverse informational perspectives. The RAG drafter generates multiple candidate responses, which the final synthesis module, typically a larger language model, to ensure coherence and completeness.

While Speculative RAG improves retrieval diversity and reduces computational costs, its modular architecture adds complexity. The additional drafter model requires fine-tuning to maintain knowledge alignment with the primary language model. Moreover, effective document clustering is critical ; failure to capture key information may impact response accuracy.

In sentiment-driven chatbots, Speculative RAG enables nuanced sentiment interpretation by generating diverse responses. However, its complexity necessitates precise tuning to align sentiment-based clustering with conversational objectives. Future research could explore sentiment-aware clustering to refine its application in dialogue systems.

6.3. Retrieval-Augmented Generation Baseline selections

6.3.1. Naive RAG

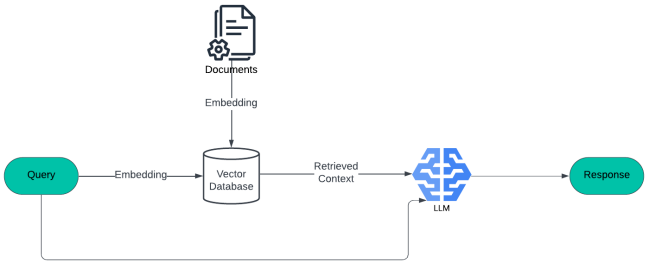

FIGURE 6 – Naive RAG Architecture.

The concept of "*naive RAG*" (see Figure 6) forms the foundational basis in the evolution of RAG systems. According to Eibich et al. [7], naive RAG is defined as a baseline system that does not incorporate advanced techniques or enhancements. It serves primarily as a benchmark for evaluating the performance of more sophisticated RAG methods, offering a simple yet essential framework for initial comparisons and development within the field.

Gao et al. [8] further describe naive RAG as the earliest methodology within the RAG research paradigm, which gained prominence shortly after the widespread adoption of ChatGPT. The naive RAG follows a traditional "Retrieve-Read" framework that involves indexing, retrieval, and generation processes. This conventional approach underscores the initial attempts to integrate retrieval mechanisms with generative models, setting a precedent for subsequent innovations and improvements.

Naive RAG, despite its foundational role, is characterized by several significant drawbacks. The precision and

recall of retrieval processes in naive RAG are often suboptimal, leading to hallucinations or incomplete responses as the system may rely on outdated or irrelevant information from its indexed sources. Additionally, the generative component of naive RAG can suffer from issues such as hallucinations, irrelevant context, and potential toxicity or bias in the model's output. Integrating context from retrieved passages can be challenging, resulting in disjointed or repetitive output. Generative models in naive RAG may overly depend on augmented information, providing limited new value or synthesized insights, thereby hindering the overall coherence and informativeness of the generated responses. Naive RAG represents the foundational efforts in RAG research, providing a starting point for more advanced and modular approaches that address its limitations [8].

6.3.2. Multi-query

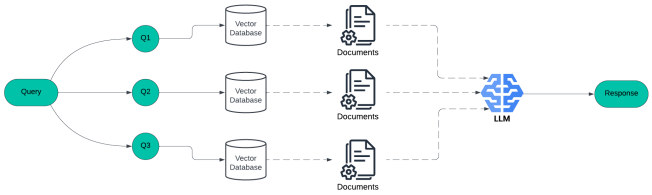

FIGURE 7 – Multi-query Architecture.

The concept of the "*multi-query*" technique (Figure 7), as described in recent academic literature, represents an advanced method for enhancing document retrieval processes. According to Eibich et al. [7], this technique involves the expansion of a single user query into multiple similar queries with the aid of a Large Language Model. This method is designed to generate alternative questions that encapsulate the intent of the original query from various perspectives, thereby broadening the scope of potential answers. Each of these queries, including the original one, is then vectorized and subjected to an individual retrieval process. This approach increases the probability of retrieving a more substantial volume of relevant information from the document repository. Subsequently, a reranker, utilizing machine learning models, is employed to filter through the retrieved chunks, prioritizing those most relevant to the initial query.

In another context, Gao et al. [8] describe the multi-query approach within the framework of RAG-Fusion. Here, the approach enhances traditional search systems by generating multiple diverse perspectives from the user's queries using an LLM. This ensures that the search results are closely aligned with both the explicit and implicit intentions of the user, leading to the discovery of more insightful and relevant information.

These descriptions underscore the multi-query technique as a pivotal advancement in the field of document retrieval, leveraging the capabilities of LLMs to expand and diversify queries, thereby improving the relevance and breadth of retrieved information.

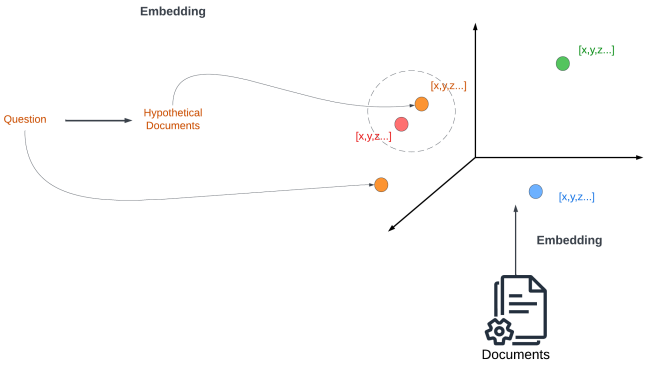

FIGURE 8 – Hypothetical Document Embedding Architecture.

6.3.3. Hypothetical Document Embedding

The "*Hypothetical Document Embedding (HyDE)*" technique (Figure 8) represents an innovative enhancement in the realm of document retrieval processes, leveraging the capabilities of LLMs. According to Eibich et al. [7] HyDE operates by generating a hypothetical answer to a query, which is richly contextual. This hypothetical answer is subsequently embedded into the vector space and employed to refine and concentrate document retrieval efforts. The core advantage of HyDE lies in the LLMs' ability to produce detailed and contextually appropriate answers, thereby aiding in the retrieval of more pertinent documents that inform the final generated response.

Gao et al. [8] further elucidate the mechanics of HyDE, describing it as a method predicated on the premise that generated answers might be more proximate in the embedding space than a direct query. The HyDE approach encompasses the following sequential steps :

1) **Creation of Hypothetical Document** : Utilizing an LLM, HyDE generates a hypothetical document (answer) in response to a query.
2) **Embedding of Hypothetical Document** : The generated hypothetical document is then embedded into the vector space.
3) **Retrieval of Real Documents** : This embedding is employed to retrieve actual documents that are similar to the hypothetical one.

The fundamental idea behind HyDE is to prioritize embedding similarity from one answer to another rather than basing similarity on the original query. However, this method may not consistently yield desirable outcomes, particularly when the language model lacks familiarity with the subject matter, potentially leading to inaccuracies [8].

Overall, the HyDE technique leverages the strength of LLMs to generate and embed hypothetical answers, refining the document retrieval process and potentially enhancing the relevance of the retrieved documents. Nonetheless, the efficacy of this approach may be contingent on the language model's domain knowledge, highlighting an area for further research and improvement.

6.4. Reranker

Retrieval-Augmented Generation reranking is a sophisticated process aimed at optimizing the selection of documents retrieved in response to a query, prioritizing the most relevant and contextually appropriate information to enhance the quality of generated responses. The reranking process typically involves several steps and employs advanced machine learning techniques to ensure the highest relevance and accuracy of the final document set.

According to Agarwal et al. [2], the reranking process begins with initial filtering, where a retriever selects the top-k potential candidates deemed most relevant based on the initial query. These candidates are then reranked using a LLM, which generates a permutation of these papers in descending order of relevance, ensuring that the generated content is grounded in the most relevant and recent research.

Kim & Min [13] emphasize the necessity for high relevance in sensitive areas, such as pharmaceutical regulatory compliance, where only highly relevant documents should be used to ensure reliability. The reranking approach involves an initial scoring agent, a high-performance LLM that quantitatively assesses the relevance of each document. The focus later shifts to using a reranking model, such as the BGE reranker, which evaluates each document chunk and assigns relevance scores to select the top-ranked documents for final response generation.

Eibich et al. [7] describe specific reranking techniques that refine the selection of documents beyond cosine similarity. For instance, the Cohere Rerank tool uses a cross-encoder architecture to jointly assess document relevance. In contrast, LLM rerankers leverage the advanced language understanding of LLMs, achieving higher accuracy but at a greater computational cost. These techniques aim to enhance RAG systems by prioritizing the most relevant and contextually appropriate information for generating responses.

Gao et al. [8] highlight the dual role of reranking as both an optimizer and a refiner, providing effective and accurate input for subsequent language model processing. This process involves contextual compression, reducing document content and filtering the entire set to present the most relevant information. Various frameworks, such as LlamaIndex, LangChain, and HayStack, implement reranking using different strategies like "*Diversity Ranker*" and "*LostInTheMiddleRanker*" to optimize the reranking process.

Glass et al. [10] explain that reranking applies more computationally demanding models after merging results from various retrieval methods, such as BM25 and DPR, to refine the top passage selection. This method uses a sequence-pair classification approach, where a BERT transformer applies cross-attention over the tokens of both the query and passage, contrasting with the initial retrieval phase that employs independent representation vectors. This approach balances accuracy and scalability, ensuring effective retrieval and relevance in the generated responses.

Overall, RAG reranking is an essential process that enhances the efficiency, relevance, and contextual importance of retrieved documents. By employing advanced machine learning algorithms and specific reranking techniques, this process significantly improves the input quality for generative models, leading to more accurate and contextually rich outputs.

7. Discussion

The application of large language models (LLMs) and Retrieval-Augmented Generation (RAG) frameworks has sparked significant advancements in healthcare, especially within mental health support. LLMs, such as OpenAI's GPT series, have transformed the ability of conversational AI to process and generate human-like responses, a crucial step in enhancing patient interaction. The scalability and linguistic sophistication of LLMs enable these models to perform complex tasks, from processing patient data to generating responses that mimic human empathy and understanding, which is especially beneficial in mental health contexts.

In healthcare, NLP and NLU facilitate patient interaction, unstructured data extraction, predictive modeling, and public health initiatives. By converting complex medical data into actionable insights, LLMs contribute to improved patient outcomes and healthcare efficiency. However, challenges arise in medical misinformation risks, where AI-driven suggestions might inadvertently contradict established medical guidelines, potentially causing harm if not carefully monitored. This limitation is mainly concerning in mental health, where sensitive patient interactions demand high accuracy and empathy.

The RAG framework, addressing limitations in LLMs, enhances response accuracy by retrieving relevant information from up-to-date sources, effectively grounding the generated outputs in factual knowledge. RAG's retrieval component maintains information from extensive datasets, reducing the tendency for LLMs to "hallucinate" or provide unreliable information. This enhancement is especially valuable in mental healthcare applications, where contextual accuracy can support more personalized and sensitive responses, essential for effective mental health support.

Moreover, techniques like Hypothetical Document Embedding (HyDE) and multi-query retrieval within the RAG framework improve the relevancy and depth of information retrieved, making it an optimal tool for knowledge-intensive tasks. In mental health, RAG-powered chatbots could retrieve specific therapeutic or diagnostic information, augmenting traditional mental health support mechanisms. The integration of advanced RAG types, such as Adaptive RAG and Speculative RAG, further refines retrieval processes, optimizing for varying complexities in patient inquiries, which is particularly beneficial for mental health contexts requiring nuanced responses.

The integration of RAG in mental healthcare applications presents both opportunities and challenges. While RAG enhances LLMs by retrieving relevant information from external databases, its effectiveness is constrained by privacy concerns surrounding patient data. The sensitivity of mental health records limits the availability of comprehensive and diverse datasets, thereby restricting the development

of a robust knowledge base for improved retrieval and response generation. This limitation may lead to incomplete or biased outputs, potentially affecting the reliability of sentiment analysis in mental health conversations. Future research should explore privacy-preserving techniques, such as differential privacy or federated learning, to facilitate secure data access while maintaining ethical standards in mental healthcare applications.

Lastly, the synergy between LLMs and RAG has transformative potential for healthcare, mainly in mental health support, by ensuring accurate, contextually rich, and empathetic patient interactions. As these technologies evolve, they are poised to address current limitations, bringing increased reliability and effectiveness to AI-driven healthcare solutions.

8. Conclusion

In conclusion, the advancements in Natural Language Processing, Natural Language Understanding, and Retrieval-Augmented Generation frameworks have brought transformative impacts to the healthcare sector, particularly in the realm of conversational AI systems. These technologies have enhanced the ability to process unstructured data, extract meaningful insights, and provide real-time support through chatbots, thereby improving both patient care and healthcare system efficiency. The exploration of RAG displays the critical role of integrating external knowledge sources to enhance the factual accuracy of language models. Despite existing challenges, such as the potential for misinformation in healthcare applications, the ongoing progress in language model development, alongside techniques like Hypothetical Document Embeddings (HyDE) and multi-query retrieval, offers promising solutions. As these technologies continue to evolve, they are likely to play an increasingly significant role in enhancing healthcare outcomes and broadening the applications of conversational AI across diverse sectors.

Références

[1] Eleni Adamopoulou and Lefteris Moussiades. An overview of chatbot technology. In *IFIP international conference on artificial intelligence applications and innovations*, pages 373–383. Springer, 2020.

[2] Shubham Agarwal, Issam H. Laradji, Laurent Charlin, and Christopher Pal. Litllm : A toolkit for scientific literature review, 2024.

[3] Akari Asai, Zeqiu Wu, Yizhong Wang, Avirup Sil, and Hannaneh Hajishirzi. Self-rag : Learning to retrieve, generate, and critique through self-reflection, 2023.

[4] Guendalina Caldarini, Sardar Jaf, and Kenneth McGarry. A literature survey of recent advances in chatbots. *Information*, 13(1) :41, 2022.

[5] Zhibo Chu, Shiwen Ni, Zichong Wang, Xi Feng, Chengming Li, Xiping Hu, Ruifeng Xu, Min Yang, and Wenbin Zhang. History, development, and principles of large language models-an introductory survey. *arXiv preprint arXiv :2402.06853*, 2024.

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. Bert : Pre-training of deep bidirectional transformers for language understanding. arxiv. *arXiv preprint arXiv :1810.04805*, 2019.

[7] Matouš Eibich, Shivay Nagpal, and Alexander Fred-Ojala. Aragog : Advanced rag output grading, 2024.

[8] Yunfan Gao, Yun Xiong, Xinyu Gao, Kangxiang Jia, Jinliu Pan, Yuxi Bi, Yi Dai, Jiawei Sun, Meng Wang, and Haofen Wang. Retrieval-augmented generation for large language models : A survey, 2024.

[9] Yunfan Gao, Yun Xiong, Meng Wang, and Haofen Wang. Modular rag : Transforming rag systems into lego-like reconfigurable frameworks, 2024.

[10] Michael Glass, Gaetano Rossiello, Md Faisal Mahbub Chowdhury, Ankita Rajaram Naik, Pengshan Cai, and Alfio Gliozzo. Re2g : Retrieve, rerank, generate, 2022.

[11] Soyeong Jeong, Jinheon Baek, Sukmin Cho, Sung Ju Hwang, and Jong C. Park. Adaptive-rag : Learning to adapt retrieval-augmented large language models through question complexity, 2024.

[12] Zhengbao Jiang, Frank F Xu, Luyu Gao, Zhiqing Sun, Qian Liu, Jane Dwivedi-Yu, Yiming Yang, Jamie Callan, and Graham Neubig. Active retrieval augmented generation. *arXiv preprint arXiv :2305.06983*, 2023.

[13] Jaewoong Kim and Moohong Min. From rag to qa-rag : Integrating generative ai for pharmaceutical regulatory compliance process, 2024.

[14] Patrick Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive nlp tasks, 2021.

[15] Huayang Li, Yixuan Su, Deng Cai, Yan Wang, and Lemao Liu. A survey on retrieval-augmented text generation. *arXiv preprint arXiv :2202.01110*, 2022.

[16] Irene Li, Jessica Pan, Jeremy Goldwasser, Neha Verma, Wai Pan Wong, Muhammed Yavuz Nuzumlalı, Benjamin Rosand, Yixin Li, Matthew Zhang, David Chang, et al. Neural natural language processing for unstructured data in electronic health records : a review. *Computer Science Review*, 46 :100511, 2022.

[17] Saskia Locke, Anthony Bashall, Sarah Al-Adely, John Moore, Anthony Wilson, and Gareth B Kitchen. Natural language processing in medicine : a review. *Trends in Anaesthesia and Critical Care*, 38 :4–9, 2021.

[18] Abu Kaisar Mohammad Masum, Sheikh Abujar, Sharmin Akter, Nushrat Jahan Ria, and Syed Akhter Hossain. Transformer based bengali chatbot using general knowledge dataset. In *2021 20th IEEE International Conference on Machine Learning and Applications (ICMLA)*, pages 1235–1238. IEEE, 2021.

[19] Rade Matic, Milos Kabiljo, Miodrag Zivkovic, and Milan Cabarkapa. Extensible chatbot architecture using metamodels of natural language understanding. *Electronics*, 10(18) :2300, 2021.

[20] Shervin Minaee, Tomas Mikolov, Narjes Nikzad, Meysam Chenaghlu, Richard Socher, Xavier Amatriain, and Jianfeng Gao. Large language models : A survey, 2024.

[21] Humza Naveed, Asad Ullah Khan, Shi Qiu, Muhammad Saqib, Saeed Anwar, Muhammad Usman, Naveed Akhtar, Nick Barnes, and Ajmal Mian. A comprehensive overview of large language models, 2024.

[22] Boci Peng, Yun Zhu, Yongchao Liu, Xiaohe Bo, Haizhou Shi, Chuntao Hong, Yan Zhang, and Siliang Tang. Graph retrieval-augmented generation : A survey, 2024.

[23] Zackary Rackauckas. Rag-fusion : A new take on retrieval augmented generation. *International Journal on Natural Language Computing*, 13(1) :37–47, February 2024.

[24] Stuart J Russell and Peter Norvig. *Artificial intelligence : a modern approach*. Pearson, 2016.

[25] Jonathan P Scaccia and Victoria C Scott. 5335 days of implementation science : using natural language processing to examine publication trends and topics. *Implementation Science*, 16(1) :47, 2021.

[26] Abonia Sojasingarayar. Seq2seq ai chatbot with attention mechanism, 2020.

[27] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need.(nips), 2017. *arXiv preprint arXiv :1706.03762*, 10 :S0140525X16001837, 2017.

[28] Haifeng Wang, Jiwei Li, Hua Wu, Eduard Hovy, and Yu Sun. Pre-trained language models and their applications. *Engineering*, 25 :51–65, 2023.

[29] Zilong Wang, Zifeng Wang, Long Le, Huaixiu Steven Zheng, Swaroop Mishra, Vincent Perot, Yuwei Zhang, Anush Mattapalli, Ankur Taly, Jingbo Shang, Chen-Yu Lee, and Tomas Pfister. Speculative rag : Enhancing retrieval augmented generation through drafting, 2024.

[30] Shi-Qi Yan, Jia-Chen Gu, Yun Zhu, and Zhen-Hua Ling. Corrective retrieval augmented generation, 2024.

[31] Wayne Xin Zhao, Kun Zhou, Junyi Li, Tianyi Tang, Xiaolei Wang, Yupeng Hou, Yingqian Min, Beichen Zhang, Junjie Zhang, Zican Dong, Yifan Du, Chen Yang, Yushuo Chen, Zhipeng Chen, Jinhao Jiang, Ruiyang Ren, Yifan Li, Xinyu Tang, Zikang Liu, Peiyu Liu, Jian-Yun Nie, and Ji-Rong Wen. A survey of large language models, 2023.

CHAPTER II

ARTICLE 2: SENTIMENTCAREBOT: RETRIEVAL-AUGMENTED GENERATION CHATBOT FOR MENTAL HEALTH SUPPORT WITH SENTIMENT ANALYSIS

2.1 RÉSUMÉ EN FRANÇAIS DE L'ARTICLE

Le système mondial de soins de santé mentale fait face à divers défis en matière d'accessibilité et de disponibilité du soutien spécialisé, tels que les psychologues et les conseillers, notamment à la suite de la pandémie de COVID-19. Cette étude explore une solution potentielle à ce problème en développant un modèle de chatbot, *SentimentCare-Bot*, qui intègre l'analyse des sentiments avec des techniques de la génération augmentée de récupération (RAG) et des modèles de langage avancés (LLMs). L'étude utilise un ensemble de données publiques de «Mental Health Counseling Conversations »et des méthodes de sélection de bases telles que «Naive RAG », «Multi-query RAG »et «Hypothetical Document Embeddings »(HyDE) pour améliorer les traductions de requêtes. Les résultats du test de «Tukey's Honest Significant Difference »(HSD) révèlent une amélioration significative des performances de l'analyse des sentiments lorsqu'elle est appliquée au «Multi-query RAG »utilisant le modèle de langage MistralAI, comparé au «Multi-query RAG »utilisant le modèle de langage d'OpenAI et à HyDE utilisant OpenAI avec l'analyse des sentiments. Ces résultats démontrent le potentiel de l'analyse des sentiments pour améliorer l'efficacité des chatbots de santé mentale.

2.2 SENTIMENTCAREBOT: RETRIEVAL-AUGMENTED GENERATION CHATBOT FOR MENTAL HEALTH SUPPORT WITH SENTIMENT ANALYSIS

Available online at www.sciencedirect.com

ScienceDirect

Procedia Computer Science 251 (2024) 334–341

Procedia Computer Science

www.elsevier.com/locate/procedia

The 14th International Conference on Current and Future Trends of Information and Communication Technologies in Healthcare (ICTH 2024)
October 28-30, 2024, Leuven, Belgium

SentimentCareBot: Retrieval-Augmented Generation Chatbot for Mental Health Support with Sentiment Analysis

Jean Pierre Nayinzira[a,*], Mehdi Adda[a]

[a] *University of Quebec at Rimouski, 300 allée des Ursulines, Rimouski G5L 3A1, Canada*

Abstract

The global mental healthcare system faces various challenges in terms of accessibility and the availability of specialist support, such as psychologists and counselors, especially following the COVID-19 pandemic. This study explores a potential solution to this problem by developing a chatbot model, SentimentCareBot, which integrates sentiment analysis with retrieved-augmented generation (RAG) techniques and Large Language Models (LLMs). The study uses a public Mental Health Counseling Conversations Dataset and baseline selection methods such as Naive RAG, Multi-query RAG, and Hypothetical Document Embeddings (HyDE) to improve query translations. The findings from Tukey's Honest Significant Difference (HSD) test reveals a significant improvement in sentiment analysis performance when it is applied to the Multi-query RAG using the MistralAI language model, compared to both Multi-query RAG using the OpenAI language model and HyDE using OpenAI with Sentiment Analysis. These results demonstrate the potential of sentiment analysis to enhance the effectiveness of mental health chatbots.

Keywords: Mental Health; Chatbot; Retrieval-Augmented Generation (RAG); Sentiment Analysis; Large Language Models (LLMs).

1. Introduction

The World Health Organization (WHO) brings attention to the critical importance of mental health as a fundamental human well-being, by promoting the improvement of global mental health. They focus on the significant gap between the demand for mental health services and their current availability, despite different public health crises, in particular the COVID-19 pandemic. WHO recommends integrating mental health services into primary healthcare and achieving universal health coverage [24]. In addition, the WHO points out the essential role of healthcare professionals in expanding mental healthcare, recommending increased training and support networks to improve service provision. It calls on governments to boost mental health funding to facilitate these necessary changes.

* Corresponding author. Tel.: +1-418-509-7757.
E-mail address: JeanPierre.Nayinzira@uqar.ca

1877-0509
10.1016/j.procs.2024.11.118

In a related study, Abd-Alrazaq et al. [1] explore the potential of chatbots in mental healthcare, noting their effectiveness in addressing disorders such as depression and stress. Although acknowledging the need for improvements in chatbot conversation capabilities and integration into healthcare, the study highlights patient perceptions and opinions, pointing to chatbots as a cost-effective and accessible complement to traditional mental health treatments.

While chatbots have proven effective in mental health contexts, several challenges remain. One key issue is the ethical concern about the reliability and effectiveness of these chatbots, many of which lack evidence-based support or sufficient research backing. It is crucial for the development and deployment of mental health chatbots to be anchored in clinical evidence to confirm their effectiveness [10]. In addition, a significant limitation of chatbots in mental health care is their fundamental inability to experience and convey empathy as humans do. Although some users find chatbots, such as Woebot, to be empathetic and supportive, this perception is not universal. The experience of empathy from chatbots can differ significantly among individuals [5].

In this study, we introduce a chatbot named "SentimentCareBot" by implementing Retrieval-Augmented Generation (RAG) models across different baseline selections (section 3.4) and then applying sentiment analysis to all of these models. To evaluate these models, we used two datasets: a complete evaluation dataset comprising 106 question-answer pairs, and a subset of this dataset consisting of five question-answer pairs. For evaluation, we used Ragas evaluation metrics to calculate *Faithfulness*, *Answer Relevancy*, and *Answer Correctness* scores for the various RAG models. Additionally, we selected two Large Language Models (LLMs) for our evaluation: OpenAI's "gpt-3.5-turbo-0125" [23] and Mistral's "mistral-large-latest" [21]. Finally, we performed an ANOVA and Tukey HSD test to analyze the significant differences between various RAG models, applying sentiment analysis to each scenario. Our results also included an analysis of token usage and latency for both LLMs, using the subset evaluation data to demonstrate the API's limitations on the study.

2. Background

With the proposal of the Turing test by Alan Turing in 1950, which posed the question "Can machines think?", the foundation for chatbot development was laid, signaling the start of their popularization. The first notable chatbot, Eliza, developed in 1966, mimicked a psychotherapist by employing simple pattern matching and template-based responses to echo user input as questions. Despite its rudimentary conversational abilities, Eliza managed to confuse people during an era when computer interaction was rare, thus encouraging the creation of more sophisticated chatbots. Significant advancements was achieved with the introduction of PARRY in 1972, a chatbot designed with a distinct personality. Later, the 1995 creation of the chatbot ALICE, which was awarded the "most human computer" title after winning the Loebner Prize in 2000, 2001, and 2004, showcased further progress. ALICE used a pattern matching algorithm enhanced by the Artificial Intelligence Markup Language (AIML), facilitating the expansion of its knowledge base by developers [2].

Machine learning and natural language processing advancements have facilitated the development of advanced chatbots, including Amazon's Alexa and Google's Assistant, reducing reliance on rules and pattern matching. These innovations have improved chatbot flexibility, ease of implementation, and the precision of human-like conversation, presenting clear advantages over rule-based models in terms of adaptability and domain independence. Furthermore, machine learning enables chatbots to learn directly from human dialogue, eliminating the need to manually define patterns, and thus increasing their adaptability [6].

Within the framework of Seq2Seq (Sequence-to-Sequence) models, the attention mechanism plays a crucial role in overcoming the limitations posed by encoding entire input sentences into fixed-length context vectors. This innovative mechanism enables the decoder to selectively focus on various parts of the input sequence, effectively preserving vital information and context that could be lost in longer sequences. By forming direct connections between the target and source, the attention mechanism allows the model to concentrate on relevant segments of the input during translation or response generation, thereby enhancing the model's capability to handle variable-length sequences and improving the simplicity and clarity of generated responses. Such advancements make the attention mechanism a crucial element in the development of more sophisticated and human-like conversational agents.

Further insights on the evolution of chatbot technologies are transformer models that employ multiple attention mechanisms and demonstrate better performance and accuracy over traditional Seq2Seq models equipped with attention mechanisms. The key to the transformer model's success lies in its ability to address sequence-related challenges

without relying on recurrent neural networks (RNNs), thus optimizing training time and enhancing neural machine translation performance. Nuanced handling and interpretation of input sequences facilitated by the attention mechanisms within transformers result in more accurate and contextually relevant responses. This leap forward marks a significant step in the ongoing effort to harness advanced neural network techniques, including Long-Short-Term Memory (LSTM) cells [20, 25].

The Transformer Architecture, which employs a self-attention mechanism to process complex data sequences, is central to the development of LLMs. This design facilitates parallel computations and effectively manages long-distance dependencies in text, laying the groundwork for the Generative Pre-trained Transformer (GPT) series. Notably, these models excel in generating text that mirrors human quality and achieve high accuracy in various linguistic tasks, thanks to extensive training on broad textual corpora [8].

Despite these technological strides, integrating LLMs into healthcare presents distinct challenges, particularly medical misinformation. Concerns include the misinterpretation of medical terms and the generation of advice that contradicts standard medical practices. Furthermore, the use of commercial servers by LLMs to store patient health data has sparked debates about privacy implications [26].

Addressing the limitations of LLMs, the RAG framework offers a distinct approach by tackling the challenges of updating knowledge bases, clarifying prediction rationales, and correcting factual inaccuracies. By combining a pre-trained sequence-to-sequence model with a detailed context index, RAG ensures more accurate and fact-based context generation. This innovative method not only reduces the likelihood of misinformation, but also enhances the interpretability and adaptability of the model, setting a new standard for knowledge-intensive natural language processing tasks [18, 15, 19, 14].

3. Method

3.1. Retrieval-Augmented Generation (RAG)

Retrieval-Augmented Generation is an advanced method designed to enhance the factual correctness and relevance of generated content by integrating external knowledge from retrieved documents into the generation process. This approach is crucial to address the inherent limitations of LLMs, such as hallucinations and outdated knowledge, thus based the generated output on current factual information from external sources [3].

The RAG framework employs retrieval mechanisms to obtain a list of relevant documents that provide contextual knowledge to support the generation process. This integration is intended to reduce the likelihood of hallucinations and update the information with the latest research [3, 16, 11]. The typical RAG workflow includes three essential steps: corpus partitioning and vector indexing, identifying and retrieving chunks based on vector similarity, and synthesizing a response conditioned on the retrieved chunks [14].

RAG combines both the innate knowledge of LLMs and additional information fetched from external documents to generate more accurate and contextually rich responses. This dual reliance on expansive background documents and generative capabilities ensures comprehensive responses to user queries, particularly in knowledge-dense NLP tasks. As a result, RAG often outperforms conventional sequence-to-sequence (seq2seq) models and certain retrieve-and-extract architectures [16].

3.2. SentimentCareBot Architecture overview

Figure 1 illustrates the proposed model for the SentimentCareBot architecture. This model comprises a baseline selection through query translation and a vector database that simplifies retrieval through similarity search. Subsequently, a sentiment analysis ranker was used to filter the retrieved documents based on their relevance and sentiment score. Finally, the LLM uses the re-ranked documents, alongside the initial input query, to generate a final response.

3.3. Data Preparation

In this study, we used the ”Mental Health Counseling Conversations Dataset,” accessible through Hugging Face [4]. This dataset consists of 3,512 collections of questions and answers sourced from two online counseling and

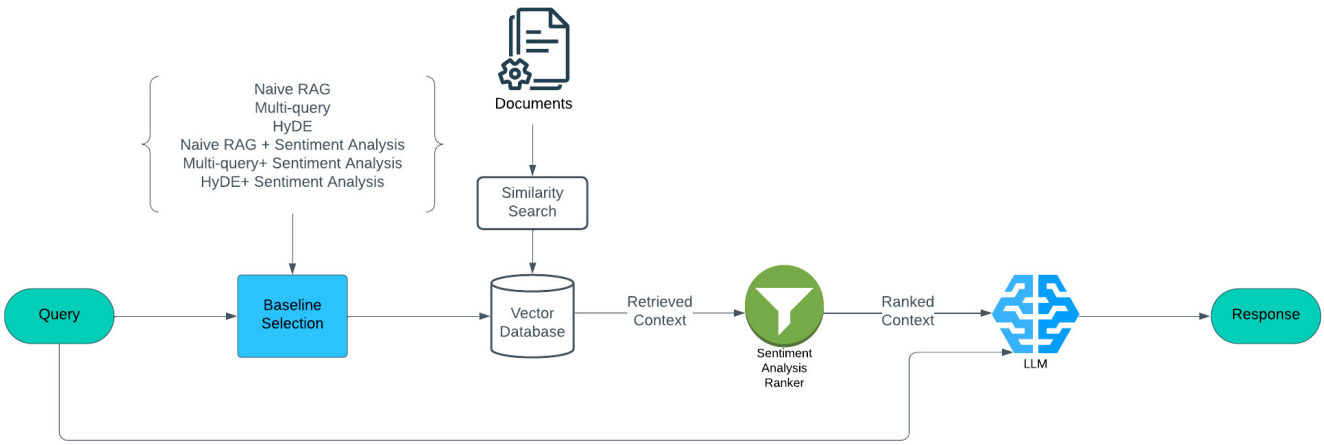

Fig. 1: SentimentCareBot Architecture

therapy platforms. The dataset provides answers to the questions that cover a wide range of mental health topics. We started by splitting the documents into chunks. This process involves dividing large documents into smaller and more manageable sections to minimize information loss. Next, we created a vector database to store and retrieve vectors. These vectors are lists of numbers that represent data within a multi-dimensional space. Using vector search methods, it is possible to locate similar data by querying for the nearest vectors in this space. Finally, a similarity search retrieves relevant documents from the database. We used Facebook AI Similarity Search (FAISS) as our similarity search metric. FAISS is a specialized library for efficient similarity search and clustering of dense vectors [9, 22].

3.4. Baseline Selection

For our baseline selection, we used Naive RAG, a conventional RAG model that simply employs the original user question for document retrieval. The Multi-query approach expands a single user query into multiple similar queries with the assistance of an LLM by generating several alternative questions that reflect the intent of the original query from different perspectives [17]. And Hypothetical Document Embeddings (HyDE), which generates a hypothetical document based on the query using LLMs [13].

3.5. Sentiment Analysis

The study analyzes the significant improvement sentiment analysis can make in the different RAG models, with the retrieval process for documents with better relevance to the question that forms the basis for the final answer. Although the documents retrieved by the FAISS similarity search may be relevant to the query, Sentiment Analysis can alter the priority of the retrieved documents by considering the highest sentiment score. We employed ”cardiffnlp,” a pre-trained model from Hugging Face [7], for text classification sentiment analysis. This model outputs the sentiment label and sentiment score. By applying this model, we can rank the retrieved documents based on their sentiment label, from positive sentiment to neutral sentiment to negative sentiment. After ranking the documents according to their sentiment label, those with similar sentiments are also ranked based on their highest to lowest scores.

3.6. LLM

For experimental purposes, we selected ”*gpt-3.5-turbo-0125*” and ”*mistral-large-latest*” due to their cost-effectiveness and ease of implementation. Given the token limitations established by the APIs provided by ”*OpenAI*” and ”*MistralAI*”, it was critical to regulate the number of queries used for each session.

3.7. Evaluation Data and Metrics

The evaluation data consists of the complete 106 question-answer pairs, which represents approximately 3% of the entire dataset, and the remaining 97% of the dataset was used to construct the vector database. For our evaluation, we used the LLMs-as-judges metric, called the Retrieval Augmented Generation Assessment (Ragas) [12]. Ragas evaluates RAG systems and introduces a suite of metrics that do not rely solely on ground-truth human annotations. Such as the "*Faithfulness*," which assesses the factual consistency of the generated answer relative to the given context, the "*Answer Relevancy*," which assesses how pertinent the generated answer is to the given prompt, and the "*Answer Correctness*" which evaluates the accuracy of the generated answer in comparison to the ground truth that offers a comprehensive evaluation of your system's performance.

4. Results

The study evaluates a variety of advanced RAG models using metrics such as *Faithfulness*, *Answer Relevancy*, and *Answer Correctness*. A comparative analysis is presented through boxplots to visualize the distribution of these metrics on the complete 106 question-answer pairs and the subset of five question-answer pairs of the evaluation data. The ANOVA and Tukey HSD tests were then used to determine the statistical significance of the differences observed in both evaluation data. Then the token usage and latency of the subset of the evaluation data are demonstrated through a boxplot.

4.1. Comparative Analysis of Metrics

The faithfulness evaluation remained relatively stable for both the complete and subset evaluation. In terms of Answer Relevancy of the complete evaluation data, as depicted in Figure 2 show greater consistency across different techniques. The analysis for the subset evaluation data (Figure 3), the median remained consistent on different RAG models except for both Naive RAG and Multi-query using MistralAI language model where the sentiment analysis improved the median.

In terms of Answer Correctness of the complete evaluation (Figure 4) the median scores indicate a small improvements when sentiment analysis is applied across different RAG models. As for the subset evaluation as depicted in Figure 5 shows the effect of sentiment analysis across various RAG models, particularly in the Naive RAG model using the MistralAI language model. Here, the median score improved by 0.51. However, the analysis also revealed exceptions, such as in the HyDE model using OpenAI language model, where both the median and maximum scores decreased, indicating that sentiment analysis may not universally benefit all architectures. Additionally, outliers in several models indicate performance variability, which may reflect scenarios where the models either under-performed or achieved exceptionally high correctness scores.

4.2. Statistical Validation of Differences

Table 1: Tukey's HSD test results comparing different RAG models.

RAG Model	Comparison	Meandiff	P-adj	Reject
Naive RAG + OpenAI	Naive RAG + Sentiment Analysis + OpenAI	0.0287	0.9992	False
Multi-query + OpenAI	Multi-query + Sentiment Analysis + OpenAI	-0.0304	0.9987	False
Multi-query + OpenAI	Multi-query + Sentiment Analysis + Mistral	-0.1103	0.0333	True
HyDE + OpenAI	HyDE + Sentiment Analysis + OpenAI	0.0081	1	False
HyDE + Sentiment Analysis + OpenAI	Multi-query + Sentiment Analysis + Mistral	-0.1085	0.0394	True

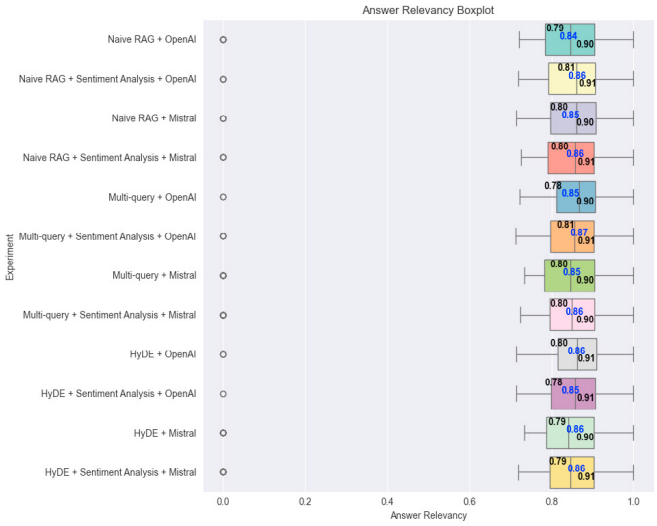

Fig. 2: Boxplot of Answer Relevancy illustrating the range distribution of Answer Relevancy scores across different RAG models.

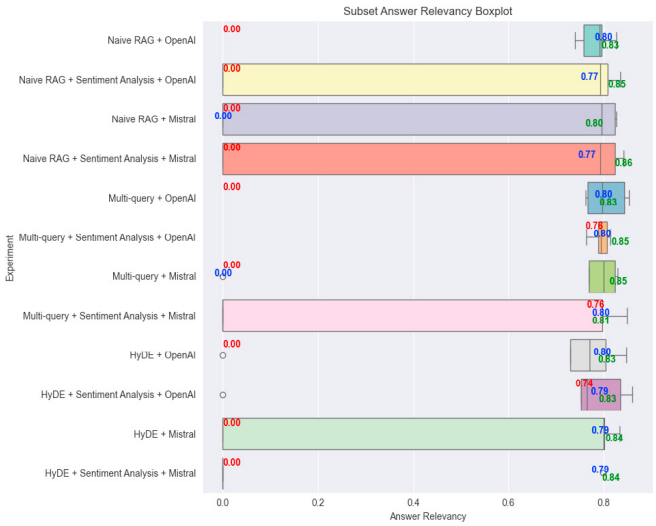

Fig. 3: Boxplot of Answer Relevancy illustrating the range distribution of Answer Relevancy scores across different RAG models on the subset evaluation data.

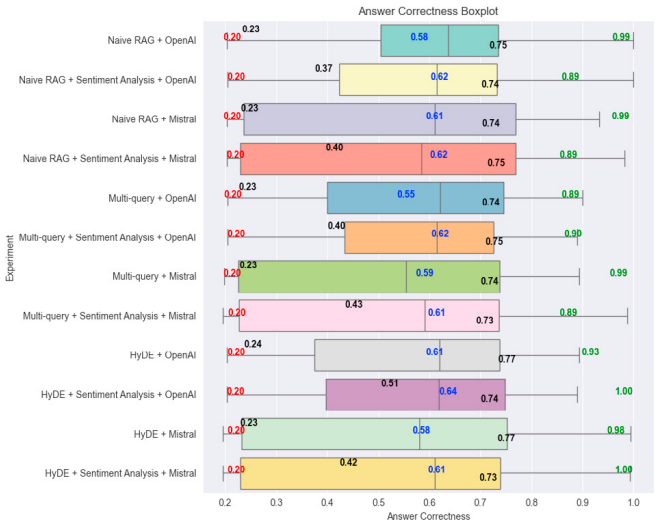

Fig. 4: Boxplot of Answer Correctness illustrating the range distribution of Answer Correctness scores across different RAG models.

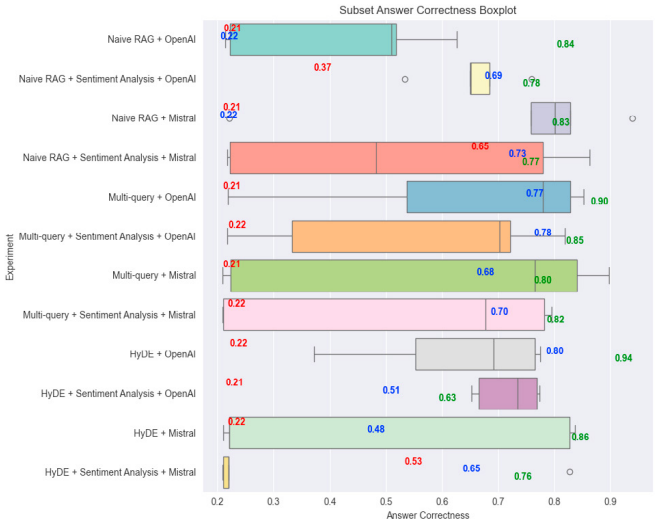

Fig. 5: Boxplot of Answer Correctness illustrating the range distribution of Answer Correctness scores across different RAG models on a subset evaluation data.

The ANOVA test we used on the evaluation results showed that there was a difference between the Answer Relevancy result groups across different RAG models of the complete evaluation data. The results of Tukey's HSD test confirmed that the Multi-query using the MistralAI language model, combined with sentiment analysis, performs better than both the Multi-query using OpenAI and the HyDE using OpenAI combined with sentiment analysis. However, other comparisons did not show a significant difference.

4.3. API performance Analysis

We evaluated the API's performance using the subset evaluation data, that consisted of five question-answer pairs. The API's token limitation constraints led us to choose this subset. The evaluation used Ragas evaluation metrics and

Langsmith[1] to examine token usage and latency for each model. By employing a manageable subset that all the RAG models could handle simultaneously, this enabled us to correctly assess the model performance.

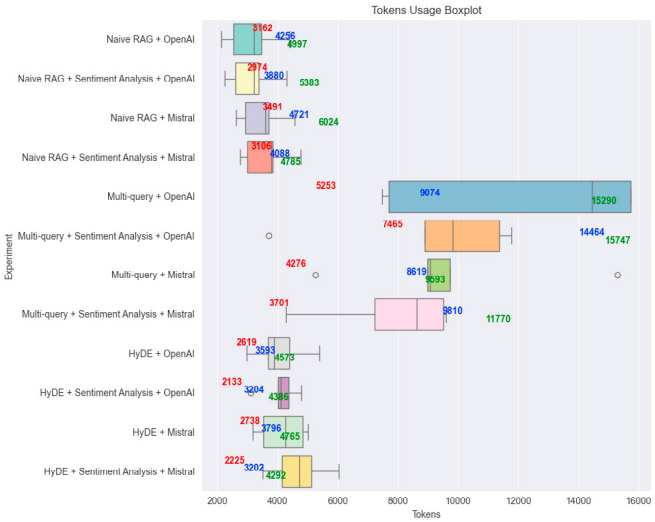

Fig. 6: Boxplot illustrating Tokens usage across different RAG models on a subset evaluation data.

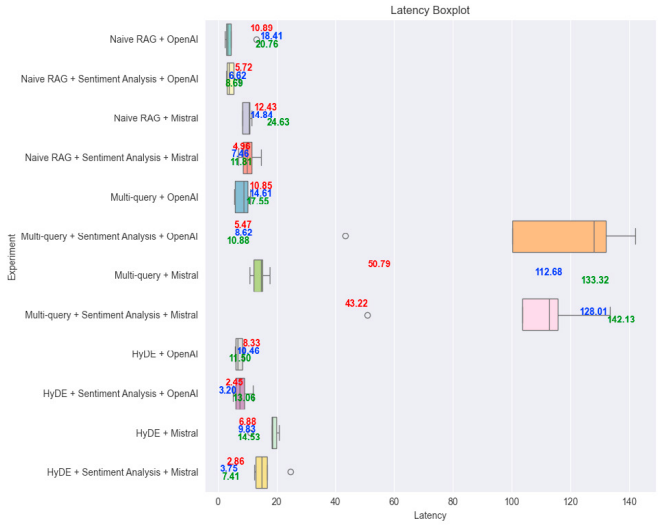

Fig. 7: Boxplot illustrating Latency across different RAG models on a subset evaluation data.

The token usage (Fig. 6) demonstrates the disparity between the performance of various RAG models. The Multi-query model using OpenAI language model demonstrated the highest token usage, with the median token count increasing when Sentiment Analysis was applied. In contrast, the HyDE baseline selection displayed the lowest token usage among the baseline selections. Despite the high token usage observed in the Multi-query baseline selection with OpenAI language model, the latency (Fig. 7) remains quite low. This indicates an efficient processing capability despite the increased token load. However, the Multi-query baseline selection using MistralAI language model displayed a higher latency, suggesting that while it may handle complex queries, it does so at the expense of response time.

5. Conclusion

In conclusion, adding sentiment analysis to the Retrieval-Augmented Generation architectures has shown a lot of promise for making mental health chatbots more useful, even though there are some challenges and restrictions associated with it. Among the tested configurations, the Multi-query approach with MistralAI's language model and sentiment analysis significantly outperforms the Multi-query method with OpenAI's language model and the HyDE approach with OpenAI's language model combined with sentiment analysis. Nevertheless, this enhanced performance comes at the cost of increased latency and token consumption, which raises concerns regarding the scalability and efficiency of API's usage. Additionally, the "*gpt-3.5-turbo-0125*" model supports a maximum of 16,384 tokens per session, while the "*mistral-large-latest*" version allows for up to 32,748 tokens. This disparity reveals critical constraints in our evaluation process, as these token limits restrict the ability to process larger datasets simultaneously. Furthermore, privacy concerns surrounding patient data contribute largely to the scarcity of publicly available mental health conversation datasets, another critical limitation. Overcoming these challenges is critical for advancing the use of sentiment analysis within RAG models in mental healthcare systems.

[1] https://smith.langchain.com

Acknowledgement

We acknowledge the support of the Natural Sciences and Engineering Research Council of Canada (NSERC) [funding reference number 06351]).

References

[1] Abd-Alrazaq, A.A., Alajlani, M., Ali, N., Denecke, K., Bewick, B.M., Househ, M., 2021. Perceptions and opinions of patients about mental health chatbots: scoping review. Journal of medical Internet research 23, e17828.
[2] Adamopoulou, E., Moussiades, L., 2020. An overview of chatbot technology, in: IFIP international conference on artificial intelligence applications and innovations, Springer. pp. 373–383.
[3] Agarwal, S., Laradji, I.H., Charlin, L., Pal, C., 2024. Litllm: A toolkit for scientific literature review. URL: https://arxiv.org/abs/2402.01788, arXiv:2402.01788.
[4] Bertagnolli, N., 2020. Counsel chat: Bootstrapping high-quality therapy data. URL: https://huggingface.co/datasets/Amod/mental_health_counseling_conversations.
[5] Boucher, E.M., Harake, N.R., Ward, H.E., Stoeckl, S.E., Vargas, J., Minkel, J., Parks, A.C., Zilca, R., 2021. Artificially intelligent chatbots in digital mental health interventions: a review. Expert Review of Medical Devices 18, 37–49.
[6] Caldarini, G., Jaf, S., McGarry, K., 2022. A literature survey of recent advances in chatbots. Information 13, 41.
[7] Camacho-Collados, J., Rezaee, K., Riahi, T., Ushio, A., Loureiro, D., Antypas, D., Boisson, J., Espinosa-Anke, L., Liu, F., Martínez-Cámara, E., et al., 2022. Tweetnlp: Cutting-edge natural language processing for social media. arXiv preprint arXiv:2206.14774 .
[8] Chu, Z., Ni, S., Wang, Z., Feng, X., Li, C., Hu, X., Xu, R., Yang, M., Zhang, W., 2024. History, development, and principles of large language models-an introductory survey. arXiv preprint arXiv:2402.06853 .
[9] Danopoulos, D., Kachris, C., Soudris, D., 2019. Approximate similarity search with faiss framework using fpgas on the cloud, in: International Conference on Embedded Computer Systems, Springer. pp. 373–386.
[10] Denecke, K., Abd-Alrazaq, A., Househ, M., 2021. Artificial intelligence for chatbots in mental health: opportunities and challenges. Multiple perspectives on artificial intelligence in healthcare: Opportunities and challenges , 115–128.
[11] Eibich, M., Nagpal, S., Fred-Ojala, A., 2024. Aragog: Advanced rag output grading. URL: https://arxiv.org/abs/2404.01037, arXiv:2404.01037.
[12] Es, S., James, J., Espinosa-Anke, L., Schockaert, S., 2023. Ragas: Automated evaluation of retrieval augmented generation. arXiv preprint arXiv:2309.15217 .
[13] Gao, L., Ma, X., Lin, J., Callan, J., 2022. Precise zero-shot dense retrieval without relevance labels. arXiv preprint arXiv:2212.10496 .
[14] Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., Dai, Y., Sun, J., Wang, M., Wang, H., 2024. Retrieval-augmented generation for large language models: A survey. URL: https://arxiv.org/abs/2312.10997, arXiv:2312.10997.
[15] Jiang, Z., Xu, F.F., Gao, L., Sun, Z., Liu, Q., Dwivedi-Yu, J., Yang, Y., Callan, J., Neubig, G., 2023. Active retrieval augmented generation. arXiv preprint arXiv:2305.06983 .
[16] Kim, J., Min, M., 2024. From rag to qa-rag: Integrating generative ai for pharmaceutical regulatory compliance process. URL: https://arxiv.org/abs/2402.01717, arXiv:2402.01717.
[17] Langchain, 2023. Query transformations. https://blog.langchain.dev/query-transformations/ .
[18] Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., tau Yih, W., Rocktäschel, T., Riedel, S., Kiela, D., 2021. Retrieval-augmented generation for knowledge-intensive nlp tasks. arXiv:2005.11401.
[19] Li, H., Su, Y., Cai, D., Wang, Y., Liu, L., 2022. A survey on retrieval-augmented text generation. arXiv preprint arXiv:2202.01110 .
[20] Masum, A.K.M., Abujar, S., Akter, S., Ria, N.J., Hossain, S.A., 2021. Transformer based bengali chatbot using general knowledge dataset, in: 2021 20th IEEE International Conference on Machine Learning and Applications (ICMLA), IEEE. pp. 1235–1238.
[21] MistralAI, 2024. mistral-large-latest [large language model]. URL: https://docs.mistral.ai/getting-started/models/.
[22] Mu, C., Yang, B., Yan, Z., 2019. An empirical comparison of faiss and fenshses for nearest neighbor search in hamming space. arXiv preprint arXiv:1906.10095 .
[23] OpenAI, 2021. gpt-3.5-turbo-0125 [large language model]. URL: https://platform.openai.com/docs/models/gpt-3-5-turbo.
[24] Organization, W.H., et al., 2022. World mental health report: Transforming mental health for all. UMB Digital Archive .
[25] Sojasingarayar, A., 2020. Seq2seq ai chatbot with attention mechanism. arXiv:2006.02767.
[26] Zhao, W.X., Zhou, K., Li, J., Tang, T., Wang, X., Hou, Y., Min, Y., Zhang, B., Zhang, J., Dong, Z., Du, Y., Yang, C., Chen, Y., Chen, Z., Jiang, J., Ren, R., Li, Y., Tang, X., Liu, Z., Liu, P., Nie, J.Y., Wen, J.R., 2023. A survey of large language models. arXiv:2303.18223.

CHAPTER III

RESULTS AND DISCUSSION

In the previous section, the interpretation of the experimentation results was limited by publication requirements. In this section, we provide a comprehensive evaluation of various advanced RAG models, by assessing their performance using key metrics such as *Faithfulness*, *Answer Relevancy*, and *Answer Correctness*. To facilitate the comparative analysis, we present boxplots that illustrate the distribution of these metrics across the complete set of 106 question-answer pairs, as well as a subset of five question-answer pairs from the evaluation dataset. In addition, we conducted ANOVA and Tukey's HSD tests to examine the statistical significance of observed differences across the models. Furthermore, we analyzed token usage and latency for the evaluation subset, using boxplots to assess API performance and limitations across different RAG models.

3.1 COMPARATIVE ANALYSIS OF METRICS

3.1.1 FAITHFULNESS

The boxplots in Figures 3.1 and 3.2 provide a detailed examination of the *Faithfulness* metric across different RAG models for both the complete and the subset evaluation datasets. Across most architectures, the median *Faithfulness* scores remain relatively stable, suggesting that the choice of RAG technique alone does not significantly enhance the overall *Faithfulness* of the models. However, some variations are notable when focusing on specific models and datasets.

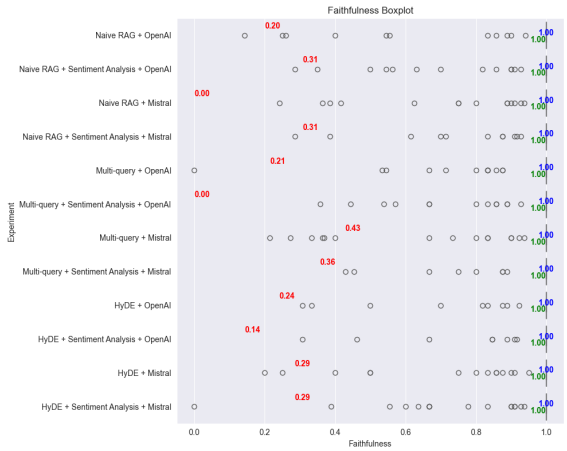

Figure 3.1 : Boxplot of *Faithfulness* illustrating the range distribution of *Faithfulness* scores across different RAG models.

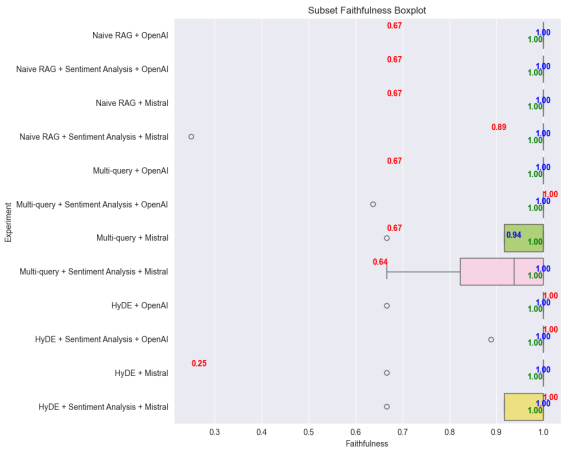

Figure 3.2 : Boxplot of *Faithfulness* illustrating the range distribution of *Faithfulness* scores across different RAG models on a subset evaluation data.

In the subset's metric scores (Figure 3.2), the Naive RAG approach combined with both MistralAI and OpenAI shows consistent minimum scores of 0.67 across various architectures. Yet, in the complete evaluation dataset (Figure 3.1), the minimum *Faithfulness* scores for Naive RAG with MistralAI are noticeably lower, particularly when Sentiment Analysis is not applied. This indicates that, while the median scores are unaffected, there is a greater occurrence of lower *Faithfulness* scores in the broader dataset, highlighting potential inconsistencies in performance.

The Multi-query and HyDE approaches, particularly when using the OpenAI model, demonstrate a higher overall *Faithfulness* in both the subset and complete evaluations. However, there is a significant drop in the minimum *Faithfulness* score when Sentiment Analysis is omitted, especially in the complete evaluation dataset (Figure 3.1). This suggests that while these approaches perform well on average, their reliability in preserving *Faithfulness* can be compromised without the additional layer of Sentiment Analysis. The HyDE approach with

MistralAI, in particular, shows a marked improvement in the minimum *Faithfulness* scores in the subset evaluation, further emphasizing the benefits of incorporating Sentiment Analysis.

Moreover, across the various methods, the application of Sentiment Analysis consistently improves the lower bounds of the *Faithfulness* metric, especially in the subset evaluation data. This improvement indicates that Sentiment Analysis is effective at reducing the occurrence of very low *Faithfulness* scores, thereby increasing the reliability of the models. Notably, however, the Naive RAG baseline with the OpenAI model does not show any change in its minimum *Faithfulness* score, even with the application of Sentiment Analysis, suggesting that this specific configuration may be less responsive to such enhancements.

Lastly, the boxplots reveal the presence of outliers in several techniques, particularly in the Multi-query approach with MistralAI, where *Faithfulness* scores shows a wide range. This irregularity indicates that while the median scores are consistent, the performance can vary significantly depending on the specific queries or subsets of data being evaluated. This outlier analysis underscores the need for caution when interpreting the median scores alone, as they may not fully capture the range of potential outcomes.

In summary, while the median *Faithfulness* scores across different RAG models do not show significant improvements, the application of Sentiment Analysis contributes to enhancing the reliability of the models by improving the lower bounds of *Faithfulness*, particularly in specific architectures. Outliers suggest variability in performance, highlighting the importance of considering both central tendencies and the entire distribution of results when evaluating model effectiveness.

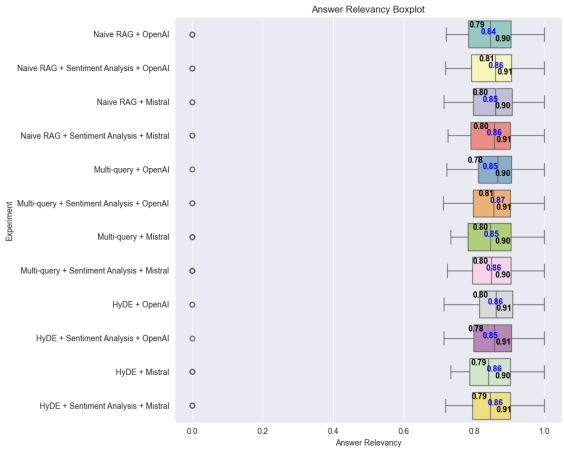

Figure 3.3 : Boxplot of *Answer Relevancy* illustrating the range distribution of *Answer Relevancy* scores across different RAG models.

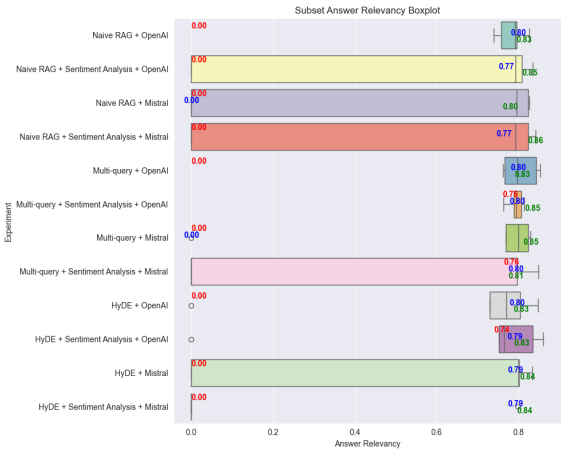

Figure 3.4 : Boxplot of *Answer Relevancy* illustrating the range distribution of *Answer Relevancy* scores across different RAG models on a subset evaluation data.

3.1.2 ANSWER RELEVANCY

The analysis of *Answer Relevancy*, as depicted in the boxplots for both the complete and the subset evaluation data (Figures 3.3 and 3.4), reveals that the median values across various RAG techniques generally do not show significant improvement. However, a noteworthy exception is observed in the subset evaluation data, where the median score improves when employing the Multi-query approach with the MistralAI model. This suggests that certain architectures may yield better performance in specific contexts, although the overall effect on the median across the complete dataset remains limited.

The application of Sentiment Analysis further enhances the results, particularly for Naive RAG and Multi-query Baseline methods when combined with the MistralAI language model. This enhancement is reflected not only in the median scores but also in the consistency of the results, as indicated by the tighter interquartile ranges (IQR) in the boxplots. The

presence of outliers, especially in the subset evaluation data, points to some variability in performance, particularly with Naive RAG architectures. These outliers suggest that while some architectures may perform well overall, they might still produce occasional low-relevancy responses.

Moreover, specific architectures, such as the combination of Multi-query with Sentiment Analysis and MistralAI, demonstrate improved median scores and higher minimum relevancy scores within the subset evaluation. This indicates that these architectures can generate highly relevant answers more consistently. However, it is essential to note that these improvements are more pronounced in the subset evaluation data, which may limit the validity of these findings.

When examining the complete evaluation data, the results show greater consistency across different techniques, with fewer outliers and tighter IQRs, particularly for the Multi-query methods with OpenAI and MistralAI. This consistency suggests that while the subset evaluation data reveals areas of potential improvement, the broader analysis indicates that these RAG models perform more reliably when applied across the entire dataset. Consequently, while the observed improvements in the subset evaluation are promising, they should be interpreted with caution, as they may not fully translate to general use cases. Further validation across the complete dataset is necessary to confirm these findings.

3.1.3 ANSWER CORRECTNESS

The evaluation of *Answer Correctness*, as depicted in Figure 3.5, provides a comprehensive view of the impact of sentiment analysis across various RAG models. The median scores indicate substantial improvements when sentiment analysis is applied, alongside the variation in LLMs used. Notably, the interquartile range (IQR) at the 25th percentile of the

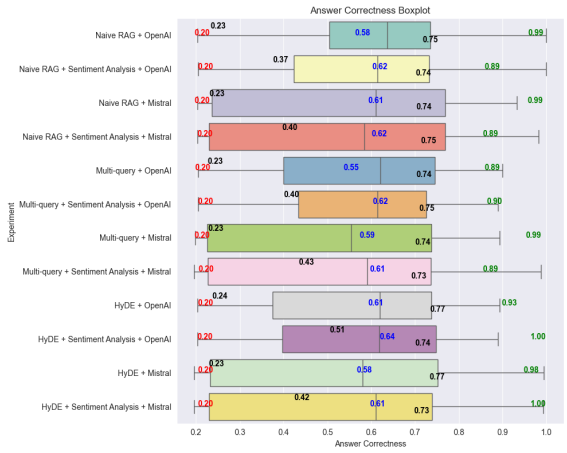

Figure 3.5 : Boxplot of *Answer Correctness* illustrating the range distribution of *Answer Correctness* scores across different RAG models.

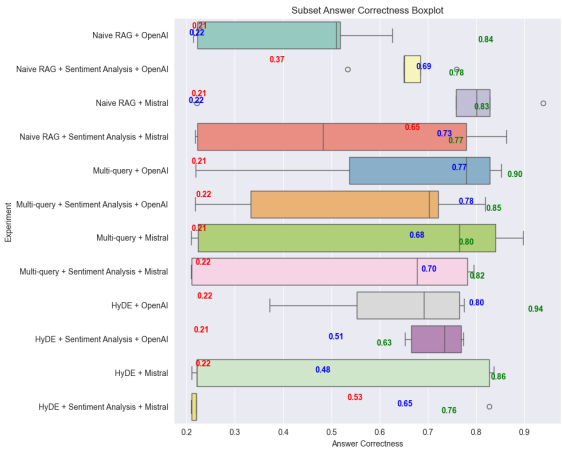

Figure 3.6 : Boxplot of *Answer Correctness* illustrating the range distribution of *Answer Correctness* scores across different RAG models on a subset evaluation data.

metric score increased by approximately 0.14 to 0.27, underscoring the enhanced robustness and consistency of the models when sentiment analysis is integrated. The range of correctness scores across different experiments also highlights significant variability, with some models demonstrating a broader distribution of scores. For instance, certain architectures, such as Naive RAG using MistralAI, exhibited maximum scores approaching 0.99, signaling exceptional performance in these cases.

Furthermore, the subset evaluation data illustrated in Figure 3.6 reinforces the positive impact of sentiment analysis across various RAG models, particularly in the Naive RAG model using the MistralAI language model. Here, the median score improved by 0.51 when sentiment analysis was applied, marking a significant enhancement. However, the analysis also revealed exceptions, such as in the HyDE model using OpenAI language model, where both the median and maximum scores decreased, suggesting that sentiment analysis may not universally benefit all architectures. Additionally, outliers in several models indicate

performance variability, which may reflect scenarios where the models either under-performed or achieved exceptionally high correctness scores. These findings emphasize that while sentiment analysis generally boosts performance, its effectiveness can vary depending on the model architecture and the criteria used for evaluation, highlighting the need for careful consideration in its application.

3.2 STATISTICAL VALIDATION OF DIFFERENCES

The ANOVA test was used on the evaluation results, and showed that there was a difference between the *Answer Relevancy* result groups across different RAG models of the complete evaluation data. Then we performed Tukey's honestly significant difference test (Tukey's HSD) to evaluate the statistical differences across various configurations to assess the influence of sentiment analysis and the choice of language model on the overall performance of the RAG models. The evaluation is performed by grouping the configurations (e.g., Group 1 and Group 2), where each group represents one of the two configurations being compared in pairwise tests. The mean difference (*meandiff*) value reflects the variation in average performance between the two configurations under comparison. The adjusted p-value (*p-adj*) represents the p-value after accounting for multiple comparisons, where a lower p-value indicates a statistically significant difference between the groups. The reject (*reject*) value is a boolean indicator of whether the null hypothesis (suggesting no difference between the groups) is rejected. A TRUE reject value implies a significant difference between the two configurations, whereas a FALSE value suggests no significant difference.

3.2.1 NAIVE RAG

The Tukey's HSD test results for Naive RAG groups as shown in Table 3.1. Across all the pairwise comparisons, the p-values are consistently high, and the null hypothesis is not

rejected for any of the tests. This suggests that there is no significant performance difference between the tested configurations, whether sentiment analysis is added or not, and regardless of whether MistralAI or OpenAI language models are used.

Table 3.1 : Tukey's HSD test results comparing different Naive RAG models.

RAG Model	Comparison	Meandiff	P-adj	Reject
Naive RAG + Mistral	Naive RAG + OpenAI	-0.0304	0.9987	False
Naive RAG + Mistral	Naive RAG + Sentiment Analysis + Mistral	-0.0045	1	False
Naive RAG + Mistral	Naive RAG + Sentiment Analysis + OpenAI	-0.0017	1	False
Naive RAG + OpenAI	Naive RAG + Sentiment Analysis + Mistral	0.0259	0.9997	False
Naive RAG + OpenAI	Naive RAG + Sentiment Analysis + OpenAI	0.0287	0.9992	False
Naive RAG + Sentiment Analysis + Mistral	Naive RAG + Sentiment Analysis + OpenAI	0.0028	1	False

3.2.2 MULTI-QUERY

The Table 3.2 presents a post-hoc analysis comparing different model configurations using Multi-query RAG groups. The results suggest that the addition of sentiment analysis does not consistently improve model performance across configurations. The only significant difference observed is the comparison between OpenAI language model and MistralAI paired with Sentiment Analysis revealed a *meandiff* of -0.1103, with a *p-value* of 0.0333, which was below the conventional threshold for significance, thus leading to a statistically significant difference between these two groups.

Table 3.2 : Tukey's HSD test results comparing different Multi-query RAG models.

RAG Model	Comparison	Meandiff	P-adj	Reject
Multi-query + Mistral	Multi-query + OpenAI	0.0921	0.1638	False
Multi-query + Mistral	Multi-query + Sentiment Analysis + Mistral	-0.0182	1	False
Multi-query + Mistral	Multi-query + Sentiment Analysis + OpenAI	0.0617	0.7569	False
Multi-query + OpenAI	**Multi-query + Sentiment Analysis + Mistral**	**-0.1103**	**0.0333**	**True**
Multi-query + OpenAI	Multi-query + Sentiment Analysis + OpenAI	-0.0304	0.9987	False
Multi-query + Sentiment Analysis + Mistral	Multi-query + Sentiment Analysis + OpenAI	0.0799	0.363	False

3.2.3 HYDE

The Table 3.3 presented offers an analysis of various configurations of the HyDE groups comparison. The results consistently indicate non-significant differences across all comparisons, as indicated by the high p-adj values (all above 0.05) and the corresponding "*reject*" value being FALSE.

Table 3.3 : Tukey's HSD test results comparing different HyDE RAG models.

RAG Model	Comparison	Meandiff	P-adj	Reject
HyDE + Mistral	HyDE + OpenAI	0.0601	0.787	False
HyDE + Mistral	HyDE + Sentiment Analysis + Mistral	-0.0039	1	False
HyDE + Mistral	HyDE + Sentiment Analysis + OpenAI	0.0682	0.62	False
HyDE + OpenAI	HyDE + Sentiment Analysis + Mistral	-0.0639	0.7115	False
HyDE + OpenAI	HyDE + Sentiment Analysis + OpenAI	0.0081	1	False
HyDE + Sentiment Analysis + Mistral	HyDE + Sentiment Analysis + OpenAI	0.072	0.533	False

3.2.4 BASELINE MODELS DIFFERENCE ANALYSIS

The Tukey's HSD test results (see Table 3.4) between different baseline models such as Naive RAG, Multi-query and HyDE, despite having models with significant difference between Multi-query various models. When they are compared to the other various baseline models the p-value is significantly higher than the conventional alpha level of 0.05 indicates no statistically significant difference between various groups. However, one notable exception is the comparison between HyDE using Sentiment Analysis paired with OpenAI language model and Multi-query using Sentiment Analysis paired with MistralAI language model, where the p-value is 0.0394, and the hypothesis is rejected (TRUE). This indicates a statistically significant difference between these two configurations. Overall, the table demonstrates that while there are various small mean differences between models, these are generally not significant, implying that adding sentiment analysis or changing the language models in most cases does not result in meaningful performance changes except for the noted comparison.

Table 3.4 : Tukey's HSD test results comparing different Baseline RAG models.

RAG Model	Comparison	Meandiff	P-adj	Reject
Multi-query + Mistral	Naive RAG + Mistral	0.0669	0.6489	False
Multi-query + Mistral	Naive RAG + OpenAI	0.0365	0.9935	False
Multi-query + Mistral	Naive RAG + Sentiment Analysis + Mistral	0.0624	0.7431	False
Multi-query + Mistral	Naive RAG + Sentiment Analysis + OpenAI	0.0652	0.6849	False
Multi-query + OpenAI	Naive RAG + Mistral	-0.0252	0.9998	False
Multi-query + OpenAI	Naive RAG + OpenAI	-0.0556	0.8611	False
Multi-query + OpenAI	Naive RAG + Sentiment Analysis + Mistral	-0.0297	0.999	False
Multi-query + OpenAI	Naive RAG + Sentiment Analysis + OpenAI	-0.0269	0.9996	False
Multi-query + Sentiment Analysis + Mistral	Naive RAG + Mistral	0.085	0.2665	False
Multi-query + Sentiment Analysis + Mistral	Naive RAG + OpenAI	0.0547	0.8737	False
Multi-query + Sentiment Analysis + Mistral	Naive RAG + Sentiment Analysis + Mistral	0.0806	0.349	False
Multi-query + Sentiment Analysis + Mistral	Naive RAG + Sentiment Analysis + OpenAI	0.0834	0.2956	False
Multi-query + Sentiment Analysis + OpenAI	Naive RAG + Mistral	0.0052	1	False
Multi-query + Sentiment Analysis + OpenAI	Naive RAG + OpenAI	-0.0252	0.9998	False
Multi-query + Sentiment Analysis + OpenAI	Naive RAG + Sentiment Analysis + Mistral	0.0007	1	False
Multi-query + Sentiment Analysis + OpenAI	Naive RAG + Sentiment Analysis + OpenAI	0.0035	1	False
HyDE + Mistral	Naive RAG + Mistral	0.0447	0.9675	False
HyDE + Mistral	Naive RAG + OpenAI	0.0143	1	False
HyDE + Mistral	Naive RAG + Sentiment Analysis + Mistral	0.0402	0.9856	False
HyDE + Mistral	Naive RAG + Sentiment Analysis + OpenAI	0.043	0.9756	False
HyDE + OpenAI	Naive RAG + Mistral	-0.0154	1	False
HyDE + OpenAI	Naive RAG + OpenAI	-0.0458	0.961	False
HyDE + OpenAI	Naive RAG + Sentiment Analysis + Mistral	-0.0199	1	False
HyDE + OpenAI	Naive RAG + Sentiment Analysis + OpenAI	-0.0171	1	False
HyDE + Sentiment Analysis + Mistral	Naive RAG + Mistral	0.0485	0.9414	False
HyDE + Sentiment Analysis + Mistral	Naive RAG + OpenAI	0.0181	1	False
HyDE + Sentiment Analysis + Mistral	Naive RAG + Sentiment Analysis + Mistral	0.044	0.9708	False
HyDE + Sentiment Analysis + Mistral	Naive RAG + Sentiment Analysis + OpenAI	0.0468	0.954	False
HyDE + Sentiment Analysis + OpenAI	Naive RAG + Mistral	-0.0235	0.9999	False
HyDE + Sentiment Analysis + OpenAI	Naive RAG + OpenAI	-0.0539	0.8847	False
HyDE + Sentiment Analysis + OpenAI	Naive RAG + Sentiment Analysis + Mistral	-0.028	0.9994	False
HyDE + Sentiment Analysis + OpenAI	Naive RAG + Sentiment Analysis + OpenAI	-0.0252	0.9998	False
HyDE + Mistral	Multi-query + Mistral	-0.0222	0.9999	False
HyDE + Mistral	Multi-query + OpenAI	0.0699	0.5814	False
HyDE + Mistral	Multi-query + Sentiment Analysis + Mistral	-0.0404	0.985	False
HyDE + Mistral	Multi-query + Sentiment Analysis + OpenAI	0.0395	0.9875	False
HyDE + OpenAI	Multi-query + Mistral	-0.0823	0.3161	False
HyDE + OpenAI	Multi-query + OpenAI	0.0098	1	False
HyDE + OpenAI	Multi-query + Sentiment Analysis + Mistral	-0.1005	0.0831	False
HyDE + OpenAI	Multi-query + Sentiment Analysis + OpenAI	-0.0206	1	False
HyDE + Sentiment Analysis + Mistral	Multi-query + Mistral	-0.0183	1	False
HyDE + Sentiment Analysis + Mistral	Multi-query + OpenAI	0.0737	0.4945	False
HyDE + Sentiment Analysis + Mistral	Multi-query + Sentiment Analysis + Mistral	-0.0365	0.9934	False
HyDE + Sentiment Analysis + Mistral	Multi-query + Sentiment Analysis + OpenAI	0.0433	0.9741	False
HyDE + Sentiment Analysis + OpenAI	Multi-query + Mistral	-0.0904	0.1857	False
HyDE + Sentiment Analysis + OpenAI	Multi-query + OpenAI	0.0017	1	False
HyDE + Sentiment Analysis + OpenAI	**Multi-query + Sentiment Analysis + Mistral**	**-0.1085**	**0.0394**	**True**
HyDE + Sentiment Analysis + OpenAI	Multi-query + Sentiment Analysis + OpenAI	-0.0287	0.9993	False

3.3 API PERFORMANCE ANALYSIS

We evaluated the API's performance using the subset evaluation data, that consisted of five question-answer pairs. The API's token limitation constraints led us to choose this subset.

The evaluation used Ragas evaluation metrics and Langsmith[1] to examine token usage and latency for each model. By employing a manageable subset that all the RAG models could handle simultaneously, this enabled us to correctly assess the model performance.

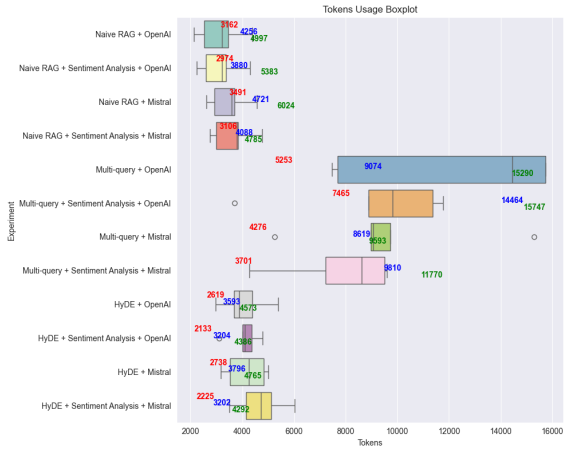

Figure 3.7 : Boxplot illustrating Tokens usage across different RAG models on a subset evaluation data.

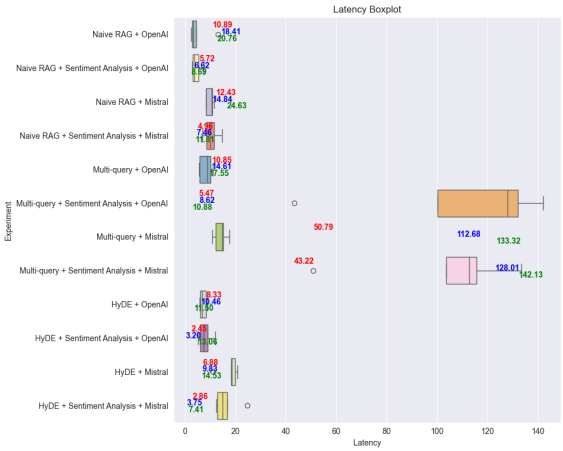

Figure 3.8 : Boxplot illustrating Latency across different RAG models on a subset evaluation data.

The token usage (Figure 3.7) demonstrates the disparity between the performance of various RAG models. The Multi-query model using OpenAI language model demonstrated the highest token usage, with the median token count increasing when Sentiment Analysis was applied. In contrast, the HyDE baseline selection displayed the lowest token usage among the baseline selections, especially when using MistralAI language model, exhibits the lowest and most consistent token usage, with minimal variation, making it a more efficient option in terms of token consumption.

In terms of latency, depicted in Figure 3.8, the performance of these models varies notably. Despite the high token usage observed in the Multi-query model with OpenAI's

[1]https://smith.langchain.com

language model, the latency remains impressively low, with a relatively narrow interquartile range, reflecting efficient processing capabilities even under a heavier token load. However, a different trend is observed with the Multi-query model using MistralAI, which shows a significantly higher median latency of around 128 seconds and a broader range, indicating less consistent performance. This suggests that while the MistralAI-powered model might be adept at handling complex queries, it does so at the expense of increased response time, highlighting a potential trade-off between managing complex operations and maintaining quick response times.

The integration of Sentiment Analysis generally results in increased token usage across the board, with the Multi-query models showing the most pronounced upward shift in median token counts and expanded usage ranges. The impact of Sentiment Analysis on latency is less marked but still present, as evidenced by slight increases in median latency values, particularly within the Multi-query model using MistralAI. These findings emphasize the importance of carefully selecting models based on the specific demands of latency and token efficiency. OpenAI's language model proves more capable of handling high token loads with lower latency, whereas MistralAI, while potentially more robust in handling complex queries, faces challenges with increased latency, particularly when additional processing tasks like Sentiment Analysis are involved.

3.4 LIMITATIONS AND PERSPECTIVES

The present study employed two LLMs for text generation tasks: OpenAI's "*gpt-3.5-turbo-0125*" and MistralAI's "*mistral-large-latest*." These models were selected primarily for their cost-effectiveness. However, each model has inherent limitations that impact their performance and evaluation. As illustrated in Figures 3.7 and 3.8, *gpt-3.5-turbo-0125* supports a maximum of 16,384 tokens per session, while *mistral-large-latest* allows up to 32,748 tokens.

This disparity shows a significant limitation in our evaluation process, as the amount of data that can be processed simultaneously is constrained by these token limits. These constraints hinder the comprehensive assessment of models using larger datasets, requiring the use of data subsets that may not fully capture the model's generalization capabilities.

Beyond token limitations, our study was constrained by the static nature of the evaluated retrieval-augmented generation (RAG) approaches. Future studies could explore agentic AI RAG systems, which incorporate reasoning and decision-making abilities to dynamically refine retrieval and generation strategies. Such approaches may better handle complex queries, contextual shifts, and multi-turn interactions, potentially enhancing real-world applicability.

Additionally, while our evaluation was conducted in a controlled setting, real-world deployment introduces additional complexities, including domain-specific nuances, user interactions, and system integration challenges. Future work should focus on deploying these models in real environments with authentic and diverse datasets to assess their adaptability, robustness, and impact in practical applications, such as mental health conversational agents. Evaluating models under real-world constraints could yield insights into their effectiveness in handling noisy data, evolving discourse, and user-specific personalization needs.

GENERAL CONCLUSION

This thesis has provided a comprehensive examination of the evolution of conversational agents, from their inception to contemporary research advancements. A central focus has been on the exploration of retrieval-augmented generation (RAG) mechanisms, highlighting various implementations and perspectives across different levels of the retrieval process. By investigating multiple RAG models, we demonstrated how sentiment analysis can be effectively integrated into mental health chatbots to enhance user interactions and response quality.

A key study within this research analyzed the impact of classifying the retrieved documents on the responses provided to users. However, this approach faced challenges, particularly regarding the limited availability of publicly accessible mental health datasets due to privacy concerns. Using the " Mental Health Counseling Conversations Dataset," our study provided strong empirical evidence supporting the positive influence of sentiment analysis, as measured by Ragas's evaluation metrics such as Answer Correctness and Answer Relevancy. A detailed evaluation of 106 question-answer pairs, including a focused subset analysis, revealed notable improvements in Answer Correctness, especially when sentiment analysis was applied to the Naïve RAG baseline model leveraging MistralAI's language model.

Furthermore, our findings highlighted performance constraints related to API-based implementations, mainly related to response latency and increased token consumption during multiple retrieval processes. Statistical analyses using ANOVA and Tukey HSD tests revealed a significant difference in performance when using the Multi-query model with MistralAI, demonstrating improved relevance when integrated with sentiment analysis. However, performance gains were accompanied by increased computational costs, posing practical limitations for real-time applications.

Despite these advancements, our research identified a critical trade-off between improved Answer Correctness and Answer Relevancy and the associated computational constraints. Token limitations imposed by APIs restrict simultaneous query processing, potentially hindering the full potential of advanced models. Moreover, privacy concerns surrounding patient data continue to limit the availability of diverse mental health conversation datasets, posing an additional challenge to further advancements in this domain.

To address these limitations, future research should explore more diverse and representative mental health conversation datasets, ensuring coverage across various demographics, linguistic styles, and clinical contexts. Additionally, leveraging advanced language models with enhanced retrieval mechanisms, such as fine-tuned transformer architectures and adaptive retrieval reranking strategies, could further refine chatbot performance. Privacy-preserving techniques, including differential privacy and federated learning, present viable solutions to mitigate data security risks while adhering to ethical and regulatory standards in mental healthcare applications. Moreover, interdisciplinary collaborations between AI researchers, clinicians, and ethicists will be crucial to ensuring that sentiment-aware RAG models are both effective and aligned with real-world therapeutic needs. By overcoming these challenges, future sentiment-enhanced RAG frameworks can be optimized to provide contextually appropriate, ethically responsible, and clinically relevant mental health support.

REFERENCES

Bertagnolli, N. (2020). *Counsel chat: Bootstrapping high-quality therapy data*. Retrieved from https://huggingface.co/datasets/Amod/mental_health_counseling_conversations

Boucher, E. M., Harake, N. R., Ward, H. E., Stoeckl, S. E., Vargas, J., Minkel, J., Parks, A. C. & Zilca, R. (2021). Artificially intelligent chatbots in digital mental health interventions: a review. *Expert Review of Medical Devices*, *18*(sup1), 37–49.

Denecke, K., Abd-Alrazaq, A. & Househ, M. (2021). Artificial intelligence for chatbots in mental health: opportunities and challenges. *Multiple perspectives on artificial intelligence in healthcare: Opportunities and challenges*, 115–128.

Es, S., James, J., Espinosa-Anke, L. & Schockaert, S. (2023). Ragas: Automated evaluation of retrieval augmented generation. *arXiv preprint arXiv:2309.15217*.

Gao, Y., Xiong, Y., Gao, X., Jia, K., Pan, J., Bi, Y., Dai, Y., Sun, J., Wang, M. & Wang, H. (2024). *Retrieval-Augmented Generation for Large Language Models: A Survey*. Retrieved from https://arxiv.org/abs/2312.10997

Jiang, Z., Xu, F. F., Gao, L., Sun, Z., Liu, Q., Dwivedi-Yu, J., Yang, Y., Callan, J. & Neubig, G. (2023). Active retrieval augmented generation. *arXiv preprint arXiv:2305.06983*.

Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Goyal, N., Küttler, H., Lewis, M., tau Yih, W., Rocktäschel, T., Riedel, S. & Kiela, D. (2021). *Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks*.

Li, H., Su, Y., Cai, D., Wang, Y. & Liu, L. (2022). A survey on retrieval-augmented text generation. *arXiv preprint arXiv:2202.01110*.

Nayinzira, J. P. & Adda, M. (2024). SentimentCareBot: Retrieval-Augmented Generation Chatbot for Mental Health Support with Sentiment Analysis. *15th International Conference on Emerging Ubiquitous Systems and Pervasive Networks / 14th International Conference on Current and Future Trends of Information and Communication Technologies in Healthcare EUSPN/ICTH 2024*, 334–341. doi: https://doi.org/10.1016/j.procs.2024.11.118

Organization, W. H. *et al.* (2022). World mental health report: Transforming mental health for all. *UMB Digital Archive*. Licence: CC BY-NC-SA 3.0 IGO.

Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L. & Polosukhin, I. (2017). Attention Is All You Need.(Nips), 2017. *arXiv preprint arXiv:1706.03762*, *10*, S0140525X16001837.

Zhao, W. X., Zhou, K., Li, J., Tang, T., Wang, X., Hou, Y., Min, Y., Zhang, B., Zhang, J., Dong, Z., Du, Y., Yang, C., Chen, Y., Chen, Z., Jiang, J., Ren, R., Li, Y., Tang, X., Liu, Z., Liu, P., Nie, J.-Y. & Wen, J.-R. (2023). *A Survey of Large Language Models*.