End-to-End Al Medical Assistant

Nisarga Deshmukh
Artificial Intelligence and Data Science and Professor. Ram Meghe Institute of Technology & Research, Amravati,444607, Maharashtra, India

***Abstract*—Artificial Intelligence (AI) and Natural Language Processing (NLP) innovations enabled the construction of a conversation machine that helps people by providing timely and useful information instantly. Against this backdrop, this paper will present the introduction of an innovative End-to-End AI Medical Assistant using the RAG approach to provide information related to health care. To ensure that it offers appropriate and credible responses to queries from customers, the system is composed of elements such as large language models, semantic search, and vector databases. Specifically, Pinecone can operate as a vector database for carrying out semantic search, and sentence-transformer embeddings can be used to vectorize the language of medicine. In addition, a local LLM provided by Ollama will be employed for generating responses. The knowledge source will include The Gale Encyclopedia of Medicine. The assistant will be delivered as a web application built using Flask. Finally, the system is aimed at providing health-related information, but never make any diagnosis – instead, it will advise seeking professional medical advice. Experimental testing reveals that the retrieval-based approach significantly improves response quality, minimizes risks associated with hallucinations, and increases system reliability.**

***Index Terms*—Artificial Intelligence, Medical Chatbot, Retrieval-Augmented Generation (RAG), Natural Language Processing, LLM, Ollama, Pinecone Vector Database, HuggingFace Embeddings, Flask Web Application.**

I. INTRODUCTION

There has been an absolute revolution in the area of information availability for the individual with the advent of AI and NLP technology, particularly in the case of healthcare. Individuals look up the web in order to find more information about diseases, its symptoms, and cure. But the information found by researching online becomes inaccurate most of the time. In addition, the medical guidance required can't be obtained easily because of the absence of healthcare experts.

As far as creating and understanding natural human languages, LLMs are highly promising, which gives an opportunity to create complex chatbots. But since these chatbots are based on existing information and may give wrong information as well, there is a need to design applications that can respond with correct information. In this regard, the retrieval augmented generation model requires the incorporation of information retrieval in the responding process.

In this research paper, we suggest an AI Medical Assistant which will provide contextually accurate information related to medicine by means of the web interface. Specifically, our assistant will fetch content from The Gale Encyclopedia of Medicine by using semantic search technology and produce the response based on the locally available language model. Combining the searching technology with the LLM technology allows us to achieve high accuracy, avoid false information, and provide safe, accurate advice to users.

II. LITERATURE REVIEW

The growth of chatbot technology has been greatly aided by developments in the area of artificial intelligence and natural language processing. Due to the introduction of Transformer architecture allowing for capturing long-range dependencies, there was a major improvement in natural language processing technologies. This Transformer architecture, which was introduced by Vaswani et al. [1] laid the foundation for today's language models. Large-scale language models like GPT [2], BERT [3] and GPT-4 [5], which use Transformer architecture have been found to perform excellently in terms of natural language comprehension and generation.

Though they function optimally, conventional language models may still generate erroneous information because they typically rely on only pre-trained knowledge. Several approaches based on Retrieval Augmented Generation (RAG) have emerged in an attempt to solve this limitation. To enhance factual correctness in knowledge-driven applications, RAG, which integrates document retrieval and text generation, was introduced by Lewis et al. [6]. Other research works such as Dense Passage Retrieval [10], REALM [9], and ColBERT [8] have continued to innovate upon document retrieval techniques through dense vector representations, hence allowing knowledge bases containing relevant documents to be retrieved and correct outputs to be provided. In addition, efficient similarity search methods such as FAISS [15] and large-scale vector searches [11], [12] have further promoted the real-world applicability of retrieval-based systems.

The creation of vector embeddings is crucial in semantic search and retrieval. An efficient way to create sentence embeddings suitable for similarity searches was proposed by Sentence-BERT [14]. Real-time infrastructure for embeddings storage and retrieval can be provided by modern vector databases such as Pinecone [13]. With the help of the discussed technologies, RAG-based pipelines may be used in real-world solutions like domain-specific chatbots.

Recently, there has been significant attention to conversational agents within the field of healthcare. Early experiments by Weizenbaum [21] showcased the potential of human-computer interactions through a pioneering conversational system called ELIZA. Conversational chatbots have been investigated more recently as a method of engaging patients and delivering healthcare services. The increasing utilization of conversational bots in healthcare applications was described in detail by Laranjo et al. [19]. In their research, Montenegro et al. [20] provided an overview of healthcare chatbots and their ability to increase the access to medical information. Reliability and minimizing risks were emphasized in works by Bickmore et al. [16] and Bibault et al. [17] that focused on the safety and clinical implications of using conversational agents in medicine.

The reliability and effectiveness of conversational agents may be enhanced significantly by integrating big language models, retrieval techniques, and vector databases, as recent studies suggest. However, the application of the Retrieval Augmented Generation approach in chatbots for medicine is still under development. In order to develop a safe and reliable AI-powered doctor's assistant, the paper builds upon previous work by implementing the three elements into one system.

Table –1: Authors, Works & Research Gaps

Authors	Work	Research Gap
Vaswani et al., 2017	Transformer architecture for NLP.	Not designed for medical domain use.
Brown et al., 2020	Large language models (GPT).	May generate incorrect medical information.
Lewis et al., 2020	Retrieval-Augmented Generation (RAG).	Not focused on healthcare applications.
Karpukhin et al., 2020	Dense Passage Retrieval.	Lacks domain-specific chatbot integration.
Reimers & Gurevych, 2019	Sentence-BERT embeddings.	No end-to-end medical chatbot use.
Bickmore et al., 2018	Safety of medical chatbots.	No retrieval-based solution proposed.
Laranjo et al., 2018	Review of healthcare chatbots.	Limited use of modern LLMs.
Montenegro et al., 2019	Survey of healthcare agents.	No RAG + vector DB integration.

III. PROBLEM STATEMENT

The majority of individuals find it difficult to obtain reliable information on their health problems. The internet is used for symptom diagnosis, illness, and remedy search, but the content is frequently false, ambiguous, and unconnected. Here, patients risk obtaining incorrect information, feeling anxious, and postponing their hospital visits. On the other hand, healthcare institutions face growing numbers of

patients and inadequate personnel to respond to patient queries.

Although AI chatbots have proven themselves to perform remarkably well in conversation, they may still generate false or misleading content. This limitation is especially critical in the healthcare sector, since false responses can be dangerous to patients' well-being. Traditional chatbots cannot be considered trustworthy enough to give real-world assistance in health matters because of the lack of verification of pre-trained knowledge based on medical literature.

Therefore, the need for a safe and dependable system that offers quick, context-related healthcare-related information, which is based on trustworthy medical literature, becomes apparent. It is challenging to develop a system that incorporates the mentioned elements without acting as an expert in medicine.

IV. PROJECT SCOPE

The development and design of an AI-based medical assistant that provides information in the domain of medicine using a conversation-driven interface are the core focus of this work. For retrieving relevant data in the form of a medical response, RAG, semantic embeddings, and vector databases have been used in the system. For ensuring real-time interaction between the user and the system, a web-based application approach has been adopted for the development of this chatbot.

The use of a dependable medical corpus ensures that this work answers informational queries about diseases, symptoms, and treatments in the realm of medicine. With a safety-first approach, this application promotes the involvement of medical professionals when required.

However, the scope of the study will only be limited to the data set selected from The Gale Encyclopedia of Medicine. Its primary objective will be to demonstrate how LLMs, vector databases, and retrieval methods can be used together for developing secure and scalable systems for storing medical data. There are several ways in which the system can be enhanced further in the future.

V. METHODOLOGY

The designed AI Medical Assistant adopts an RAG-based architecture, which consists of data pre-processing, semantic search, vector storage, prompt engineering, and deployment in real time. The entire approach includes many steps as discussed further below.

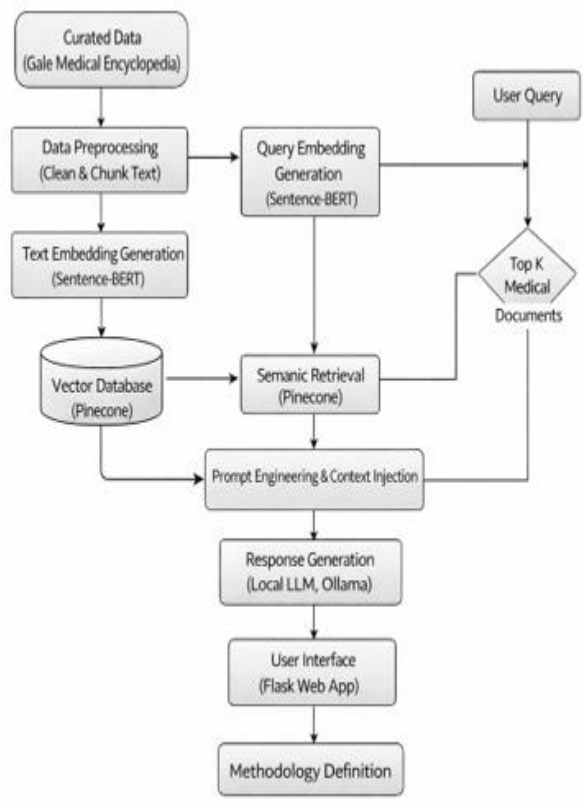

Fig-1: System Workflow

1. Data Collection and Creating Knowledge Bases

In the model, the initial process focuses on establishing a reliable database for the field of medicine. The Gale Encyclopedia of Medicine (Volumes 1 & 2) acted as the source of data collection. Instead of basing the chatbot's response on unreliable data collected from various websites, the use of a reliable source guarantees reliable medical information.

To improve the search results, the obtained information underwent pre-processing. To enhance semantic search accuracy, long texts were chunked down into shorter text segments. This process is essential as it helps the search engine find relevant context instead of lengthy information. The system was capable of finding accurate and contextually rich data because each segment was accompanied by metadata.

2. Text Embedding Creation

The sentence-transformers/all-MiniLM-L6-v2 model was applied in order to create vector representations from the chunks so that the machine will be able to understand the meaning of the literature associated with medicine. In contrast to simple text matchings, text embeddings are vectors based on the semantics of the sentences.

A semantic search algorithm needs such transformations. Thus, if the document has the word combination "symptoms of abdominal pain," what a user searches for is data on "causes of stomach pain." The embeddings allow recognizing the similarity of two sentences in semantic terms when standard keyword search would be ineffective. Matching of similarity can be guaranteed by applying the same model in indexing and queries.

3. Pinecone Vector Database Storage

The vectors were stored in Pinecone, a cloud vector database built for similarity search, after generating the embeddings. Pinecone conducts searches in the vast repository of vectors rapidly using the Approximate Nearest Neighbor algorithm (ANN).

The Pinecone index was populated with vectors from all the text fragments and their embeddings. This makes it possible to extract valuable information related to the healthcare industry in real-time fashion, despite the growing volume of data. In addition, Pinecone provides scalable, speedy, and efficient indexing that is necessary for successful real-time chatbot operation.

4. Semantic retrieval and query processing

In the first place, the system employs the same HuggingFace embedding model for embedding a user's question when it gets submitted through the chatbot interface. This ensures that both the user's queries and documents already in the database belong to the same vector space.

Subsequently, the most relevant three papers are retrieved by the Pinecone retriever with the help of a similarity search through vector similarity. Retrieving multiple documents improves response precision as well as provides extra context to the language model. This is because the LLM receives accurate information and does not have to rely only on the knowledge acquired during its training.

5. Context Injection and Prompt Engineering

The construction of a well-organized system prompt is done through the combination of the collected documents. Managing the behaviour of the language model is mainly reliant on prompt engineering.

The language model is prompted through the system prompt to:

- Use only the medical context obtained to answer the question.
- Avoid giving any diagnosis and medication prescriptions.
- Inform users if the data is insufficient.
- Guide patients toward seeking out qualified doctors.

This will help ensure the integrity of the bot.

6. Local LLM for Response Generation (Ollama)

Gemma 2B is a model that runs locally using Ollama and is applied during the process of generating responses. There are several advantages associated with local LLMs:

- Higher levels of data protection and privacy
- Reducing dependence on cloud-based APIs
- Faster response time
- Lower costs

The formatted prompt with the user's question and relevant context is fed into the LLM. It generates a coherent human-like response based on the provided medical content.

7. Implementation of an End-to-End RAG Pipeline Using LangChain

All the components were integrated into a pipeline in a single package via LangChain. The RAG workflow involves:

- Obtaining relevant documents from Pinecone
- Formatting the prompt using the context
- Notifying the LLM about the prompt
- Returning the answer generated

In order to maintain modularity in workflow management, the pipeline utilizes Runnable Lambda chains. Any future changes like that of the LLM or the vector database can be achieved without rebuilding the entire system.

8. Web Applications Implementation Based on Flask

Real-time user interactions were made possible through the implementation of the AI Medical Assistant as a web application using the Flask

framework. As it is lightweight, easy to incorporate into Python-based AI projects, and suitable for developing machine learning applications, it was selected. The software enables individuals to communicate their questions regarding their medical condition using a browser-based chat box.

The question posted by the patient is then communicated to the server using Flask technology through an HTTP request. The server processes the query and sends it to the RAG pipeline that builds a response for the user based on the medical information retrieved through the local language model. Real-time user interaction is supported by instant response transmission to the user.

This implementation makes the process scalable and secure, as the app is run locally without the need for any third-party APIs. Flask, hence, becomes the medium that connects the front end with the back end of the AI process.

VI. RESULT

With the help of RAG architecture, the implemented solution perfectly demonstrates the working of an AI Medical Assistant chatbot. For presenting the developed system, a web-based chat interface is used. It allows people to communicate via the web-based chat interface by putting forward their queries and getting instant replies to their questions related to health issues. The simplicity and ease of the system make its design user-friendly since the chat interface looks like that of any other messaging application.

The Flask backend takes the user request and then sends it to the RAG system after processing. In order to retrieve the relevant medical records, the request is turned into an embedding, which is matched against the vector database on Pinecone. The local language model running using Ollama retrieves the documents and generates an answer based on the provided medical information. Thus, it means that the bot uses information obtained from the knowledge base rather than relying only on its memory.

It becomes clear from the analysis of the output generated by the chatbot that the system understands medical terminology and generates short answers. The illness, its signs, treatment methods, and potential consequences are explained in simple language. Moreover, it changes its answer depending on the next question. It allows concluding that the system takes into consideration the context of interaction with users.

As all the mentioned components were integrated perfectly into one system, the created application functions effectively. The objectives of the project – the development of the interactive artificial intelligence medical assistant capable of obtaining information and generating meaningful insights regarding the sphere of health care – have been successfully accomplished.

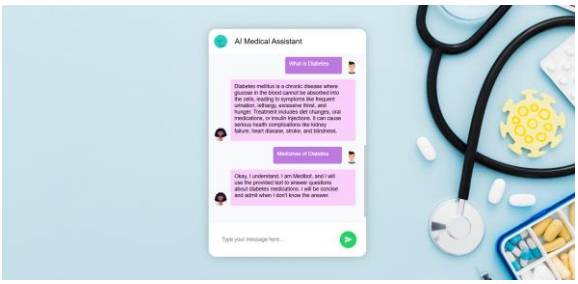

Fig -2: AI Medical Chatbot Output Interface

VII. CONCLUSION

This project applied the RAG approach to successfully create and develop an artificial intelligence-based assistant chatbot. It develops an intelligent medical question-answering application through the use of cutting-edge technologies such as vector databases, local large language models, and the web-based interface. The chatbot is able to understand customer queries, gather medical information, and respond with insightful answers within seconds.

The solution demonstrates that combining language generation and document search could be effective in achieving precision and reliability. In addition, the chatbot utilizes medical knowledge that is stored in the vector database to provide relevant answers and does not rely on the knowledge available to the language model alone. Therefore, this approach ensures that the answers are based on the retrieved information and minimizes false responses significantly.

It is equally imperative to highlight the importance of introducing AI-based systems in a simple and user-friendly manner as well. To achieve this aim, the chatbot has been designed to be easy to use and interact with using a chat-based interface with the help of the Flask web application. This fact has been proven through the successful combination of the frontend and backend.

Taking everything into consideration, it has become clear that the objective set by the project of developing an AI-based healthcare assistant capable of assisting users regarding their queries on healthcare has been achieved successfully. Through this project, the future developments of intelligent healthcare systems have been made possible.

REFERENCES

[1] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention Is All You Need," in Proc. Adv. Neural Inf. Process. Syst., 2017, pp. 5998–6008.

[2] T. Brown, B. Mann, N. Ryder, M. Subbiah, J. Kaplan, P. Dhariwal, A. Neelakantan, P. Shyam, G. Sastry, and A. Askell, "Language Models are Few-Shot Learners," in Proc. Adv. Neural Inf. Process. Syst., 2020, pp. 1877–1901.

[3] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," in Proc. NAACL-HLT, 2019, pp. 4171–4186.

[4] A. Radford, K. Narasimhan, T. Salimans, and I. Sutskever, "Improving Language Understanding by Generative Pre-Training," OpenAI, Tech. Rep., 2018.

[5] OpenAI, "GPT-4 Technical Report," arXiv:2303.08774, 2023.

[6] P. Lewis, E. Perez, A. Piktus, F. Petroni, V. Karpukhin, N. Goyal, H. Küttler, M. Lewis, W. Yih, and T. Rocktäschel, "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," in Proc. Adv. Neural Inf. Process. Syst., 2020.

[7] S. Izacard and E. Grave, "Leveraging Passage Retrieval with Generative Models for Open Domain Question Answering," in Proc. ACL, 2021, pp. 874–880.

[8] O. Khattab and M. Zaharia, "ColBERT: Efficient and Effective Passage Search via Contextualized Late Interaction," in Proc. SIGIR, 2020, pp. 39–48.

[9] J. Guu, K. Lee, Z. Tung, P. Pasupat, and M. Chang, "REALM: Retrieval-Augmented Language Model Pre-Training," in Proc. ICML, 2020, pp. 3929–3938.

[10] V. Karpukhin, B. Oguz, S. Min, P. Lewis, L. Wu, S. Edunov, D. Chen, and W. Yih, "Dense Passage Retrieval for Open-Domain Question Answering," in Proc. EMNLP, 2020, pp. 6769–6781.

[11] J. Johnson, M. Douze, and H. Jégou, "Billion-Scale Similarity Search with GPUs," IEEE Trans. Big Data, vol. 7, no. 3, pp. 535–547, 2021.

[12] M. Aumüller, E. Bernhardsson, and A. Faithfull, "ANN-Benchmarks: A Benchmarking Tool for Approximate Nearest Neighbor Algorithms," Inf. Syst., vol. 87, Jan. 2020.

[13] Pinecone Systems, "The Pinecone Vector Database," Pinecone White Paper, 2022.

[14] J. Reimers and I. Gurevych, "Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks," in Proc. EMNLP, 2019, pp. 3982–3992.

[15] Facebook AI Research, "FAISS: A Library for Efficient Similarity Search," 2020.

[16] E. Bickmore, H. Trinh, S. Olafsson, T. O'Leary, R. Asadi, N. Rickles, and R. Cruz, "Patient and Consumer Safety Risks When Using Conversational Assistants for Medical Information," npj Digital Medicine, vol. 1, no. 1, 2018.

[17] A. Bibault, T. Chaix, M. Guillemassé, S. Cousin, L. Escande, A. Perrin, B. Pienkowski, and P. Nectoux, "Healthcare Ex Machina: Are Conversational Agents Ready for Prime Time in Oncology?" Clin. Oncol., vol. 31, no. 1, pp. 17–18, 2019.

[18] S. Palanica, A. Flaschner, A. Thommandram, M. Li, and Y. Fossat, "Physicians' Perceptions of Chatbots in Healthcare," J. Med. Internet Res., vol. 21, no. 4, 2019.

[19] L. Laranjo, A. Dunn, H. Tong, A. Kocaballi, J. Chen, R. Bashir, D. Surian, B. Gallego, F. Magrabi, and A. Lau, "Conversational Agents in Healthcare: A Systematic Review," J. Am. Med. Inform. Assoc., vol. 25, no. 9, pp. 1248–1258, 2018.

[20] A. Montenegro, C. da Costa, and R. da Rosa Righi, "Survey of Conversational Agents in Healthcare," ACM Comput. Surv., vol. 51, no. 6, 2019.

[21] J. Weizenbaum, "ELIZA—A Computer Program for the Study of Natural Language Communication Between Man and Machine," Commun. ACM, vol. 9, no. 1, pp. 36–45, 1966.

[22] M. McTear, Z. Callejas, and D. Griol, The Conversational Interface: Talking to Smart Devices. Springer, 2016.
[23] M. Huang, X. Zhu, and J. Gao, "Challenges in Building Intelligent Open-Domain Dialogue Systems," ACM Trans. Inf. Syst., vol. 38, no. 3, 2020.
[24] J. Gao, M. Galley, and L. Li, "Neural Approaches to Conversational AI," Found. Trends Inf. Retr., vol. 13, no. 2–3, pp. 127–298, 2019.
[25] H. Touvron, L. Martin, K. Stone, P. Albert, A. Almahairi, Y. Babaei, and N. Bashlykov, "LLaMA: Open and Efficient Foundation Language Models," arXiv:2302.13971, 2023.