IJIREEICE ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⁑ Peer-reviewed & Refereed journal ⁑ Vol. 14, Issue 4, April 2026

DOI: 10.17148/IJIREEICE.2026.14446

CONFIDENCE-GUIDED MULTI-AGENT LLM FRAMEWORK FOR CLINICAL DECISION SUPPORT IN OPHTHALMOLOGY USING BIOMEDICAL RAG AND WEB INTELLIGENCE

Mrs .D. Archana[1], P. Tanmai Sai[2], S. Bhavana[3], Y. Suhitha Priya[4], G. Siri Saranya[5]

Assistant Professor, Department of Computer Science and Engineering (Data Science),

Andhra Loyola Institute of Engineering and Technology, Vijayawada, Andhra Pradesh, India[1]

Student, Department of Computer Science and Engineering (Data Science),

Andhra Loyola Institute of Engineering and Technology, Vijayawada, Andhra Pradesh, India[2-5]

Abstract: Eye diseases such as conjunctivitis, glaucoma, cataract, diabetic retinopathy, and optic neuritis are becoming increasingly common worldwide. If not detected at an early stage, these conditions can lead to serious vision impairment or even permanent blindness. In many parts of the world, especially in rural and remote areas, people face significant difficulty in accessing a qualified eye specialist on time. This paper presents a Confidence-Guided Multi-Agent Large Language Model (LLM) Framework developed for clinical decision support in ophthalmology. The system accepts symptom descriptions from patients written in plain English and processes them through a pipeline of five intelligent agents. Each agent performs a specific role, including symptom analysis, disease retrieval using Retrieval-Augmented Generation (RAG) with FAISS, web-based validation using DuckDuckGo, and final decision synthesis. A unique confidence scoring mechanism is incorporated, which combines the RAG similarity score and the web trust score to give users a reliable percentage-based prediction. The system is built using Python and Streamlit, making it accessible through any standard web browser without requiring specialized software. Experimental results demonstrate that the system correctly identifies common eye diseases with confidence scores above 80% for valid inputs, and appropriately rejects unrelated inputs. This system is intended for academic demonstration and can serve as a supportive tool for telemedicine platforms and rural healthcare centers.

1. INTRODUCTION

Eye diseases are among the leading causes of disability across the world. Conditions like glaucoma, conjunctivitis, cataract, diabetic retinopathy, and optic neuritis affect millions of people every year. When detected early, most of these conditions can be treated effectively. However, if ignored or delayed, they can cause permanent vision loss, which significantly impacts a person's quality of life and daily independence.

In many parts of the world, particularly in villages and smaller towns, patients do not have easy access to a qualified eye doctor. Eye care facilities with specialized equipment are mostly found in large cities, which means that rural patients either delay their doctor visits or never seek help at all. By the time they finally consult a specialist, the disease may already be at an advanced stage.

With the rapid growth of Artificial Intelligence (AI) and machine learning, it is now possible to build intelligent systems that can assist both doctors and patients. These systems can read and understand symptom descriptions, identify matching diseases from a knowledge base, and suggest the right type of specialist to visit. Such tools can save time and money, especially for patients who live far from hospitals.

In this project, we developed a Confidence-Guided Multi-Agent LLM Framework for ophthalmology clinical decision support. The system uses a pipeline of five intelligent agents, each dedicated to a specific task. One agent validates the relevance of the user input. Another agent reads and extracts key symptom terms from the typed description. A third agent searches for matching diseases using a method called Retrieval-Augmented Generation (RAG) combined with FAISS (Facebook AI Similarity Search). A fourth agent validates the result against live online medical sources or local clinical guidelines. Finally, the Decision Agent brings all the outputs together and presents the user with a clear prediction,

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering
Impact Factor 8.414 ∺ Peer-reviewed & Refereed journal ∺ Vol. 14, Issue 4, April 2026
DOI: 10.17148/IJIREEICE.2026.14446

including the disease name, a specialist recommendation, a list of associated symptoms, and a confidence score that tells the user how reliable the result is.

The system is built using Python and Streamlit, which means it runs directly in a web browser and does not require any special hardware or software. The primary goal of this project is to build something simple, affordable, and practical that can genuinely help people, especially those in areas where access to eye specialists is limited. The system is designed for academic and demonstration purposes and is not a replacement for professional medical consultation.

2. LITERATURE SURVEY

Over the last several years, many researchers have tried to apply Artificial Intelligence for medical diagnosis, particularly for eye-related diseases. Early AI systems in this domain relied on fixed rule-based approaches, where predefined conditions were used to map symptoms to diseases. For example, if a patient reported redness, the system would classify it as conjunctivitis. While these systems were straightforward to build, they lacked flexibility and could not handle variations in how patients described their symptoms.

With the rise of deep learning, Convolutional Neural Networks (CNNs) became widely adopted for the analysis of eye images, including retinal scans and fundus photographs. These image-based models achieved very good diagnostic accuracy. However, they required expensive imaging equipment and were not designed to handle symptom-based text input from patients, which limits their usability in low-resource settings.

As Natural Language Processing (NLP) matured, researchers began exploring text-based diagnostic systems. Sentence Transformer models, which convert sentences into numerical vectors, made it possible to compare user-described symptoms with a database of disease descriptions. This approach eliminated the need for imaging and made text-based diagnosis accessible on standard computers.

FAISS (Facebook AI Similarity Search) further improved this approach by enabling fast, large-scale similarity searches across thousands of disease vectors in a fraction of a second. This allowed real-time diagnosis on ordinary hardware. More recently, multi-agent architectures have been explored in medical AI to divide diagnostic responsibilities among multiple specialized agents, each responsible for a particular step. This modular approach improves accuracy, maintainability, and overall performance.

Retrieval-Augmented Generation (RAG) has also emerged as a promising technique, combining knowledge retrieval with language model reasoning to produce more accurate and explainable outputs compared to systems that rely entirely on static training data.

Despite these advancements, most existing ophthalmology AI systems fail to communicate to the user how confident the system is about its output. This gap reduces user trust and limits practical adoption. The proposed system directly addresses this issue by implementing a confidence scoring mechanism that averages the RAG similarity score and the web validation trust score to produce a final reliability percentage, giving users a clearer picture of the prediction quality.

3. PROPOSED SYSTEM

In this project, we designed a system that takes eye-related symptom descriptions from a user and returns a predicted disease along with useful clinical information. The entire process is handled by a pipeline of five intelligent agents working in a sequence, each performing a distinct and well-defined role.

The user starts by typing their symptoms in plain English using a Streamlit-based web interface. For example, they might write: “My eye is red and itchy and I have a watery discharge.” The system then processes this input step by step.

The first component in the pipeline is the Relevance Gate. This gate checks whether the user’s input is actually related to eye symptoms before any further processing takes place. If the user types something completely unrelated—such as “I have a stomach ache”—the system politely rejects it, displays a 0% confidence score, and asks the user to provide proper eye-related input. This filtering mechanism prevents the system from generating inaccurate predictions.

If the input passes the Relevance Gate, it is forwarded to the Symptom Analysis Agent. This agent cleans the text by removing unnecessary words and identifies medically significant keywords such as redness, blurred vision, itching, floaters, or watery discharge. It also assesses the severity level of the symptoms as High, Medium, or Low based on the nature and intensity of the described symptoms.

Next, the Disease Retrieval Agent converts the extracted symptoms into numerical vectors using a pre-trained Sentence Transformer model (all-MiniLM-L6-v2). These vectors are compared against a FAISS index of disease vectors to retrieve the top three most similar eye diseases, along with their similarity scores.

The top-matched disease is then passed to the Web Intelligence Agent, which validates the result by searching online medical sources using DuckDuckGo. If the internet is unavailable, it falls back to a local clinical guidelines file. The outcome of this validation step is a web trust score, which reflects how well the predicted disease is supported by external medical information.

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering
Impact Factor 8.414 ⁝ Peer-reviewed & Refereed journal ⁝ Vol. 14, Issue 4, April 2026
DOI: 10.17148/IJIREEICE.2026.14446

Finally, the Decision Agent synthesizes all outputs from the previous agents and presents the final result on the screen. This includes the predicted disease name, a recommended specialist, a list of commonly associated symptoms, the confidence score, and an explanation of how the result was derived.

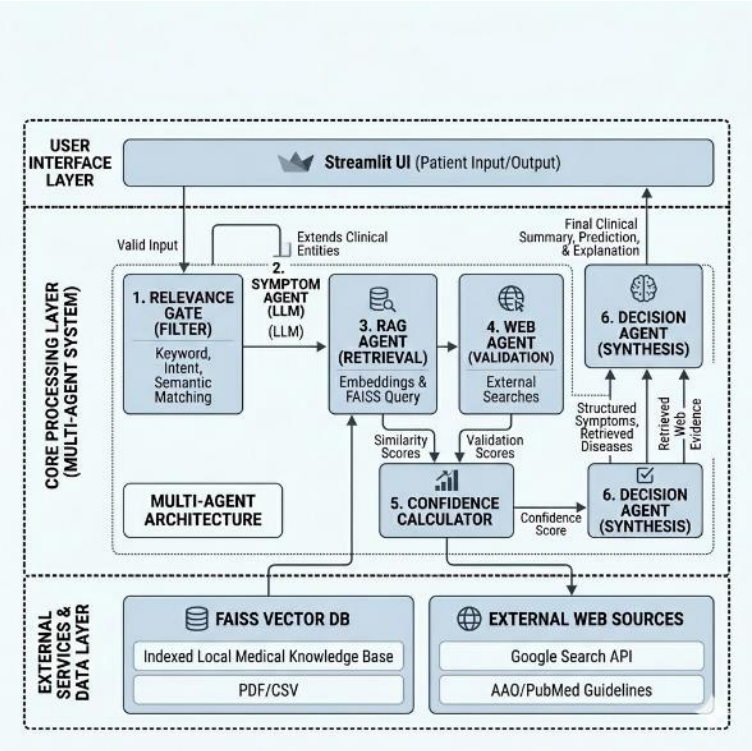

Fig 1: System Architecture of the Multi-Agent Framework

The system generates the following outputs in real time: the predicted eye disease (such as Conjunctivitis, Glaucoma, Cataract, or Diabetic Retinopathy), a confidence score expressed as a percentage, a list of extracted symptoms with their severity level, a recommended specialist (such as an Ophthalmologist or Glaucoma Specialist), and evidence sources used during validation. The design is intentionally simple and affordable, and the system runs smoothly on a regular laptop or desktop with no special hardware.

4. METHODOLOGY

The system processes the user's input through a well-defined sequence of steps. Each step performs a specific function and passes its output to the next stage in the pipeline. The complete methodology is described below.

Step 1 – Collecting Input: The user opens the Streamlit web application in their browser and types their eye symptoms in a text box. The input can be written in simple, everyday language without requiring any medical terminology.

Step 2 – Checking Relevance: Before any processing begins, the Relevance Gate checks whether the user's input contains eye-related keywords such as "eye," "vision," "redness," or "blur." If the input does not contain any relevant terms, it is immediately rejected and a 0% confidence score is displayed. This step ensures the system remains safe from generating incorrect predictions for unrelated inputs.

Step 3 – Analyzing Symptoms: The Symptom Analysis Agent cleans the text by removing unnecessary words and identifying important medical keywords. For instance, from the sentence "My eye feels gritty and looks very red," it extracts symptoms such as [gritty sensation, redness]. The agent also determines the severity level—High (if words like "pain" or "vision loss" are present), Medium (if multiple symptoms are mentioned), or Low (for mild symptoms).

Step 4 – Retrieving Diseases Using RAG and FAISS: The extracted symptoms are converted into numerical vectors using a pre-trained Sentence Transformer model (all-MiniLM-L6-v2). These vectors are compared against a FAISS index containing stored disease vectors. The top three most similar diseases are retrieved along with their cosine similarity scores.

Step 5 – Validating Results Online: The Web Intelligence Agent takes the top-matched disease and searches for it online using DuckDuckGo. It verifies whether trusted medical websites confirm that the identified disease matches the described

DOI: 10.17148/IJIREEICE.2026.14446

symptoms. In the absence of internet connectivity, a local "guidelines.txt" file is used as a fallback. The result is expressed as a web trust score (for example, 80%).

Step 6 – Calculating Confidence and Generating Output: The final confidence score is calculated as the average of the RAG similarity score and the web trust score. For example, if RAG gives 90% and Web gives 80%, the final confidence is (90 + 80) / 2 = 85%. If the confidence score falls below 60%, a warning is shown advising the user to visit a specialist. The Decision Agent then assembles and displays the full result on the Streamlit interface, including the disease name, recommended specialist, associated symptoms, confidence score, and explanation.

5. RESULTS

We tested the system with a wide variety of inputs to evaluate how well it performs across different scenarios. The results were consistently accurate for valid eye symptom inputs and appropriately safe for invalid ones.

When symptoms such as "redness, itching, gritty sensation, and watery discharge" were entered, the system correctly identified Conjunctivitis as the predicted disease and recommended an Ophthalmologist. The confidence score for this case was approximately 83%, indicating a reliable prediction.

For symptoms like "gradual blurring, halos around lights, and loss of side vision," the system correctly identified Glaucoma with a confidence score above 80% and recommended a Glaucoma Specialist for further evaluation.

When the input described "eye pain especially when moving, sudden vision loss in one eye, and flashing lights," the system identified Optic Neuritis and recommended a Neuro-Ophthalmologist. The confidence score in this case was 86.5%, reflecting the strong match between the described symptoms and the retrieved disease.

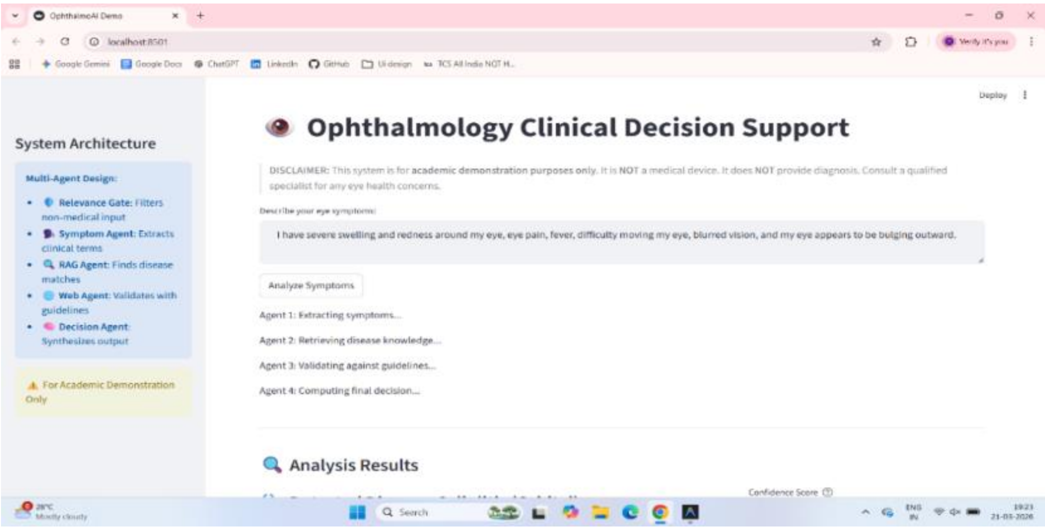

Fig 2: Streamlit Web Interface for Symptom Input and Output

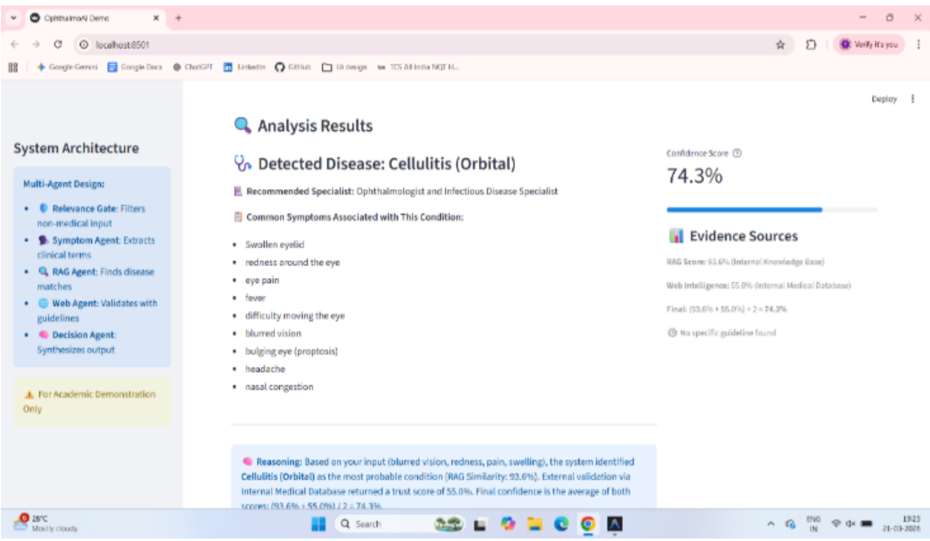

Fig 3: Confidence Score and Disease Prediction Output

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering
Impact Factor 8.414 ∺ Peer-reviewed & Refereed journal ∺ Vol. 14, Issue 4, April 2026
DOI: 10.17148/IJIREEICE.2026.14446

When the system was tested with completely unrelated input such as “I broke my leg and have a stomach ache,” the Relevance Gate correctly rejected the input, displayed a 0% confidence score, and prompted the user to enter proper eye-related symptoms. No disease prediction was made, which is the expected behavior.

Overall, the system produced correct and reliable predictions for the majority of valid inputs. It also handled invalid inputs safely by refusing to make predictions without eye-related symptoms. The confidence score mechanism made it easy for users to understand how much weight to give the system’s output, which is a particularly important feature in any medical context.

6. CONCLUSION

In this project, we successfully built an AI-based system that can assist in detecting eye diseases from patient-described symptoms. The system uses a multi-agent framework where five agents—the Relevance Gate, Symptom Analysis Agent, RAG Agent (FAISS), Web Intelligence Agent, and Decision Agent—work together in a sequential pipeline to deliver accurate and reliable results.

The use of Sentence Transformers and FAISS made disease retrieval fast and computationally lightweight. The web validation step added an extra layer of reliability by cross-checking results against actual medical sources. The confidence scoring feature helped users assess the trustworthiness of each prediction, which is a critical requirement for any medical-oriented application.

The Streamlit interface made the system easy to use for anyone, including non-technical users. A user can simply open the application in a browser, type their symptoms, and receive a result within a few seconds, without needing any prior medical knowledge or specialized software.

We believe this system can be genuinely useful in settings where access to eye specialists is limited, particularly in rural areas and telemedicine platforms. While it is not intended to replace professional medical advice, it can help individuals understand their condition earlier and seek the right kind of care faster. It is important to note that this system has been developed for academic and demonstration purposes only, and all users should consult a qualified ophthalmologist for proper diagnosis and treatment.

7. FUTURE SCOPE

There are several meaningful ways in which this system can be improved and expanded in the future. One of the most important planned enhancements is the addition of image-based diagnosis. Users could upload a photograph of their eye or a retinal scan, and the system could analyze it using computer vision techniques such as Convolutional Neural Networks. This would significantly improve the system’s usefulness for conditions like diabetic retinopathy and macular degeneration that are more accurately detected through imaging.

Expanding the disease knowledge base to include a wider range of eye conditions would also increase the system’s coverage. Currently, the system handles the most common diseases, but there are many rarer conditions that could be included with additional training data and clinical input.

Connecting the system with hospital information systems would allow doctors to view patient-generated reports directly and use the tool as a proper clinical decision support platform. A mobile application version of the system would make it accessible to a far larger audience, enabling patients in remote areas to check their symptoms from their phones and decide whether to seek immediate medical attention.

Adding multilingual support would allow users from different regions to type their symptoms in their native language, making the system more inclusive. Voice input functionality would also be beneficial for elderly users who find typing difficult.

In the long run, integrating this system with wearable health monitoring devices and deploying it on the cloud would allow it to serve a large number of users simultaneously. With these planned enhancements, the system has the potential to grow into a widely used and clinically valuable tool for early eye disease detection and management.

REFERENCES.

[1]. Li, Z., Song, D., et al., "VisionUnite: A vision-language foundation model for ophthalmology enhanced with clinical knowledge," IEEE Transactions on Pattern Analysis and Machine Intelligence, vol. 47, no. 12, 2025.

[2]. Johnson, J., Douze, M., and Jégou, H., "Billion-scale similarity search with FAISS," IEEE Transactions on Big Data, vol. 7, no. 3, pp. 535–547, 2021.

[3]. Reimers, N. and Gurevych, I., "Sentence-BERT: Sentence embeddings using Siamese BERT-networks," in Proc. EMNLP 2019, pp. 3982–3992, 2019.

ISSN (O) 2321-2004, ISSN (P) 2321-5526

International Journal of Innovative Research in Electrical, Electronics, Instrumentation and Control Engineering

Impact Factor 8.414 ⪤ Peer-reviewed & Refereed journal ⪤ Vol. 14, Issue 4, April 2026

DOI: 10.17148/IJIREEICE.2026.14446

[4]. Vaswani, A., Shazeer, N., et al., "Attention is all you need," in Advances in Neural Information Processing Systems (NeurIPS), vol. 30, 2017.

[5]. Lewis, P., Perez, E., et al., "Retrieval-augmented generation for knowledge-intensive NLP tasks," in Advances in Neural Information Processing Systems (NeurIPS), vol. 33, 2020.

[6]. Holland, R., et al., "Specialist vision-language models for clinical ophthalmology," Eye, vol. 37, pp. 1275–1283, 2023.

[7]. Zhou, Y., et al., "RETFound: A foundation model for generalizable disease detection from retinal images," Nature, vol. 620, pp. 110–118, 2023.

[8]. Kim, S., et al., "Evaluating large language models for clinical use in ophthalmology," Journal of Medical Internet Research, vol. 27, 2025.

[9]. Devlin, J., Chang, M. W., Lee, K., and Toutanova, K., "BERT: Pre-training of deep bidirectional transformers for language understanding," in Proc. NAACL-HLT, pp. 4171–4186, 2019.

[10]. Streamlit, "Open-source Python framework for building web applications," [Online]. Available: https://streamlit.io