IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 4 September 2025, accepted 19 September 2025, date of publication 23 September 2025,
date of current version 29 September 2025.

Digital Object Identifier 10.1109/ACCESS.2025.3613340

RESEARCH ARTICLE

An Adaptive Multi-Agent LLM-Based Clinical Decision Support System Integrating Biomedical RAG and Web Intelligence

ÇAĞATAY UMUT ÖĞDÜ, KÜBRA ARSLANOĞLU, (Member, IEEE), AND MEHMET KARAKÖSE, (Senior Member, IEEE)
Department of Computer Engineering, Fırat University, 23119 Elâzığ, Türkiye

Corresponding author: Kübra Arslanoğlu (karslanoglu@firat.edu.tr)

ABSTRACT Increasing data complexity in clinical decision-making processes hinders physicians' ability to make rapid and accurate decisions. This study proposes an innovative solution to this problem by designing a multi-layered, adaptive Clinical Decision Support System (CDSS) comprising interacting large language model (LLM) agents. The proposed system performs semantic-level information retrieval using a BioBERT-based vector database, enhances information retrieval by accessing up-to-date medical resources via the web, and restructures outputs by activating an adaptive optimization loop in low-confidence situations. Through the structuring of clinical texts, cross-validation of symptom analyses with literature and internet sources, and collaborative data fusion among agents, the system integrates multi-source data and produces consistent decisions. In experiments conducted on the MedQA, PubMedQA, and MedBullets datasets, the system achieved accuracies of 94%, 88%, and 84%, respectively, representing substantial improvements over state-of-the-art methods and demonstrating the significance of the proposed architecture for clinical decision-making reliability. This framework is not merely an information retrieval engine; it is a clinical intelligence partner designed to learn, actively contribute to the decision process, and focus on reliability. In contrast to current CDSS protocols, which frequently depend on static modules or single-agent models, our architecture tackles some of the shortcomings in timeliness, multi-source evidence fusion, and confidence calibration. This originality enables the system to be a next-generation clinical intelligence partner by enabling an unprecedented level of transparency, customizability, and adaptability in real-world decision-making processes.

INDEX TERMS Clinical decision support system, large language models, multi-agent system.

I. INTRODUCTION

In today's healthcare systems, the complexity of clinical data and the need for clinicians to make accurate decisions under time pressure make the need for advanced clinical decision support mechanisms even more evident [1]. Information from heterogeneous sources such as electronic health records, medical imaging, and genomic datasets significantly complicates clinical assessment processes [2]. Consequently, they also increase the risks that can negatively affect patients, such as misdiagnoses and treatment interruptions. According to a report published by the American Institute of Medicine (IOM) in 2000, approximately 44,000 to 98,000 deaths in healthcare institutions in the United States alone are associated with preventable medical errors each year. It is also emphasized that approximately 70% of these errors are human-caused [3]. While existing rule-based systems contribute to partial improvements in clinical practice, they lose their effectiveness in real-world scenarios due to their inability to adapt flexibly to dynamic patient profiles and their limitation to static algorithms [4]. Against this backdrop, the main technical challenge addressed in this study is to design a clinical decision support system that (i) ingests noisy, heterogeneous, and time-varying patient

The associate editor coordinating the review of this manuscript and approving it for publication was Qiang Li.

data; (ii) remains up-to-date with rapidly evolving medical evidence; (iii) constrains LLM hallucinations through grounding and cross-checking; and (iv) produces calibrated, explainable, and auditable recommendations under clinical time constraints.

An innovative approach to overcoming these challenges is AI-based solutions, primarily LLMs [5], [6]. Large language models such as GPT-4 and Med-PaLM have become a significant focus in medical research thanks to their capabilities in clinical data analysis, context-based analysis, and interdisciplinary data integration [7], [8]. These systems have the capacity to form the basis for a wide range of innovative applications, from patient data processing [9] to the development of dynamic treatment strategies compatible with clinical protocols [10] to the detection of fake medical images using artificial intelligence [11]. Up to this moment, systems based on structural representation of clinical knowledge have also offered important support in the decision-making processes. For instance, the ontology-based system, called "OntoDiabetic," by Sherimon and Krishnan performs risk analysis for diabetic patients by modelling the clinical guidelines using Web Ontology Language rules [12]. While such systems capture domain knowledge in a formal machine processable format to enable a logical reasoning mechanism consistently, their application is constrained by the use of static, predetermined knowledge bases. This makes it difficult to overcome the challenges associated with dynamic noisy and unstructured streams of data that characterize clinical practice and quickly changing medical evidence in the decision-making process. However, the clinical integration of LLMs faces methodological issues such as inconsistencies in the outputs they produce (e.g., hallucinations), opacity of decision-making processes, and partial incompatibility with evidence-based medicine paradigms [13]. For example, models developed by Levra and colleagues that analyze free-text notes in emergency department electronic health records are capable of distinguishing cases of syncope. While this approach demonstrates the power of LLMs in understanding implicit patterns and symptoms in unstructured clinical text, it primarily focuses on a knowledge-based recognition task within existing records [14]. Such single-model implementations cannot fully address more complex needs, such as cross-validation with external information sources (such as both current web data and custom domain RAGs) to increase the reliability of the outputs and dynamically optimizing the process in low-confidence situations. On the other hand, the inability to integrate these technologies with real-time patient data streams is one of the main obstacles currently limiting their applicability in clinical applications [15]. In [16], various large language models were tested on two separate datasets covering nine different outpatient clinic branches. In radiology trials, GPT-4 topped the list with 98% accuracy, followed closely by Llama-3.3-70b at 96%. In contrast, in some cases, measurements dropped GatorTron's accuracy by as much as 12%. The findings demonstrate that design choices such as scale, architecture, and the scope of training data used are direct determinants of success in the clinical setting.

Beyond decision support, the clinical application spectrum of LLMs is broad [7], [8]. A directly related line is the extraction of structured signals from clinical narratives, e.g., symptom identification from text [24] and patient triage [25]. In parallel, multimodal imaging applications (e.g., text-guided segmentation in echocardiography) have emerged as a complementary direction aimed at pixel-level labeling rather than end-to-end decision support, with imaging-oriented uses such as AI-based detection/verification of medical images also explored [11], [12]. In this paper, we explicitly scope our contribution to text-centric CDSS and use these neighboring lines to motivate our focus on confidence-aware, up-to-date, and verifiable recommendations.

The newly proposed KG4Diagnosis [17] study provides a remarkable answer by integrating LLMs with the processes of building knowledge graphs using a multi-agent hierarchical approach. The system, as shown in Figure 1, derives entities and relations enabled by BioBERT by breaking down medical texts into semantic chunks, translates entities into a knowledge graph, and structures the diagnostic decision-making process through task allocation between GP-LLM and expert advisor agents. While this framework is quite robust in terms of semantic integrity and structure, it falls short in being suitable for real-time streams of data, web-based up-to-date knowledge scanning, and feedback optimization with confidence score-based outputs. Taken together, existing solutions partially address accuracy or interpretability, but none simultaneously ensure timeliness (fresh evidence), robustness to heterogeneous inputs, and confidence-aware, verifiable outputs the core facets of the technical challenge.

Correspondingly, we pose the overall research question of this paper as follows: Can an LLM-based, multi-agent CDSS that simultaneously retrieves, authenticates, and aggregates biomedical RAG knowledge with web evidence in real-time and closes the loop using confidence-driven feedback deliver sound, timely, and interpretable clinical recommendations?

Our system, unlike the KG4Diagnosis architecture, aims to systematically improve the accuracy, timeliness, and explainability of clinical choices through an architecture that not only utilizes structured information but also RAG mechanisms based on dynamic web information, adaptive feedback loops, and data fusion from multiple sources. Particularly, we decompose the task into four technical sub-tasks and address each with a domain-specific mechanism: (1) robust clinical text normalization and structuring to deal with noisy inputs; (2) domain-specific biomedical retrieval (BioBERT-based RAG) to extract domain-specific terminology and abbreviations; (3) trust-aware multi-source combination with conflict detection and confidence calibration; and (4) an adaptive feedback mechanism for re-querying and hypothesis refinement when confidence is low. This decomposition makes the overall problem tractable and enables an auditable pipeline in accordance with clinical workflows.

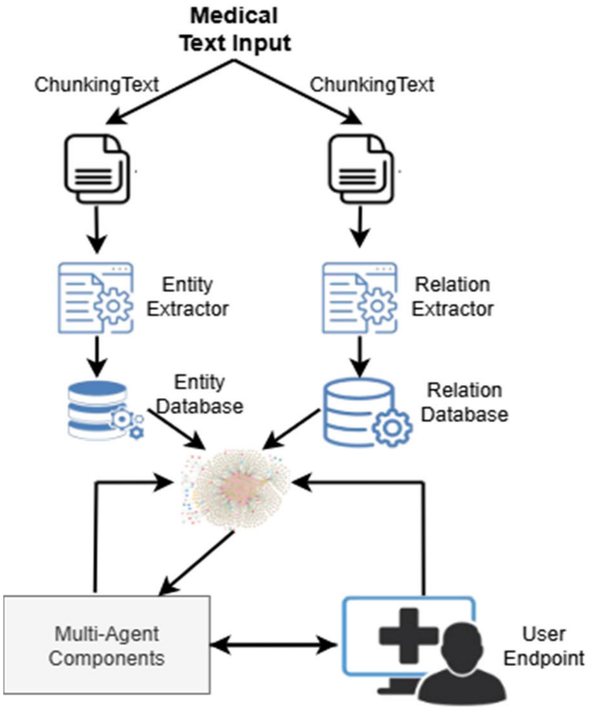


FIGURE 1. Multi-Agent supported diagnostic process flow diagram [16].

Similarly, Medical Agents [18] proposes an interactive LLM agent architecture for clinical diagnosis support based on large language models. This architecture performs symptom recognition and differential diagnosis generation by dividing tasks among expert agents with defined roles. Each agent undertakes tasks such as answering clinical questions, scoring diagnostic candidates, and analyzing symptom-diagnosis matches; thus, the decision-making process is driven by a multi-stage pool of experts. The system generates more explainable and reasoned outputs compared to traditional single-agent architectures, with the ability to compare different hypotheses across an internal representation space, analyze clinical consistency, and re-evaluate when necessary. However, because the system relies heavily on LLM content-based inference, the limited integration of external information sources (e.g., literature, web, databases) poses a constraint on accuracy and timeliness. Furthermore, due to its structure lacking dynamic adaptation or feedback loops between agents, improving decision quality largely depends on the success of a predefined task allocation. Nevertheless, it is clear that the modular structure offers a significant contribution to the field by increasing interpretability and enabling LLMs in their decision-making role.

The core capabilities and shared limitations of these pioneering approaches in the literature are presented comparatively in Table 1. In contrast, our approach explicitly targets the above technical challenge by coupling specialization (agent roles) with evidence freshness (web scanning), factual grounding (biomedical RAG), and decision calibration (confidence-guided feedback), thereby operationalizing reliability as a first-class design goal.

TABLE 1. Feature-based comparison of CDSS approaches.

Feature	Traditional LLMs [7], [8], [15]	KG4 Diagnosis [16]	Medical Agents [17]	This Study
Type	Single Model	Multi-Agent	Multi-Agent	Multi-Agent
External Info	Limited	Yes (Info Graph)	Limited	Yes (Specialized RAG)
Dynamic Web Access	No	No	No	Yes
Multi-Source Data Fusion	No	Limited	Limited	Yes
Prominent Limitation	Static information, risk of hallucinations	A structure closed to real-time data and the web	Integration of external current information is weak	-

This study does not present any truly new algorithmic paradigm, but it fills an important gap in the literature it combines existing pieces (biomedical RAG, web-based evidence retrieval, and adaptive feedback mechanisms) into a single agent-driven CDSS framework. What is original about our work is that we systematically integrate and interoperate these components in real-time within a modular, task-specific architecture. This enables us to transform approaches that have been previously tested in the literature separately into a system that is capable of providing reliable, interpretable, and dynamically optimized clinical recommendations. In particular, the actions we describe will not only enhance diagnostic accuracy but also provide a clearer and more malleable decision-making process consistent with real clinical decision-making.

This study proposes a dynamic CDSS based on an LLM-based multi-agent architecture. The synergy of modular LLM agents will provide higher diagnostic accuracy and adaptability than traditional fixed-rule systems. Unlike traditional LLMs, modular agents are designed with an architecture that combines the capabilities of (i) autonomously executing tasks aligned with clinical protocols, (ii) contextually processing real-time data streams (EHR, laboratory results, etc.), and (iii) adaptive learning with clinician feedback [19], [20]. For example, a diagnostic inference agent prioritizes potential pathologies by matching patient symptoms with EHR data and epidemiological trends, while a treatment optimization agent can generate personalized protocols by integrating patient-specific parameters (e.g., comorbidities, drug interactions) with current clinical guidelines.

Furthermore, while existing Retrieval-Augmented Generation (RAG) [22] systems, such as in [21], use general-purpose embedding models, this study fills this critical gap in the literature by using a multi-task BioBERT architecture [23] trained on biomedical NLI datasets to process medical jargon and abbreviations. BioBERT was chosen because it was pre-trained on PubMed articles. The vector database was created by combining 889 publicly available medical articles with 5,630 patient texts. While the literature has examined the performance of LLMs on isolated tasks such as symptom analysis [24] or patient triage [25], this study is the first comprehensive study to quantitatively evaluate the impact of a CDSS integrated with multi-agent interaction on clinical outcomes.

This study aims to demonstrate how LLM agents can synergize with clinicians' expertise, offering the following key contributions to the field:

1) Overcoming the limitations of fixed-rule systems, it proposes a flexible multi-agent architecture that can dynamically analyze patient data and optimize itself over time.
2) It goes beyond general-purpose solutions by offering a specialized RAG approach that ensures correct interpretation of medical terminology and abbreviations.
3) It aims to increase the reliability of the outputs produced by automatic merging and cross-checking of data from different information sources (WEB and RAG agents).

II. METHODOLOGY

This research aims to propose an integrated system architecture based on different LLM models and multi-agent interaction for the streamlining of clinical decision-making processes. The CDSS proposed here is based on four main components that incorporate a modular agent framework, dynamic data integration, contextual analysis, and adaptive learning capabilities. This framework is designed to systematically implement the key contributions highlighted in the introduction: providing a flexible architecture that addresses the limitations of fixed-rule systems, providing an understanding of storage in medical terminology with a domain-specific RAG approach, and improving durability with multi-source data persistence. The design logic of each component in the system, how it works, the methods used, the functionality of the component and its contribution to the system are discussed in detail below.

We did not perform any additional model training or fine-tuning. Because evidence is fetched at inference time, no task-specific fine-tuning is required; updates are made by re-indexing sources rather than changing model weights. All LLMs (DeepSeek-R1, Gemini-2.0-Flash, Qwen2-72B, and GPT-4o-mini) were used off-the-shelf for task-specific inference, and the biomedical sentence-transformer ('BioBERT-mnli-snli-scinli-scitail-mednli-stsb') was used only to generate embeddings for the vector database without updating weights.

A. MODULAR AGENT STRUCTURE

The schematic diagram shown in Figure 2 depicts the mechanism of cooperation among the elements of the proposed system architecture. The system consists of five domain-specialist autonomous agents that are designed to perform clinical decision support processes. The architecture is based on a multi-layered model that was developed within the CrewAI framework, with modular agents combining through sequential process flows and dynamic feedback loops. This modular and multi-layered structure forms the basis of the flexible architecture, identified as the study's first contribution, which can analyze dynamic patient data and optimize itself over time. Each agent's focus on a specific area of expertise increases the system's adaptability and flexibility in complex clinical scenarios. The "Agent, Crew, Process, Task" modules of CrewAI are used to define the tasks, abilities, and actions of the agents.

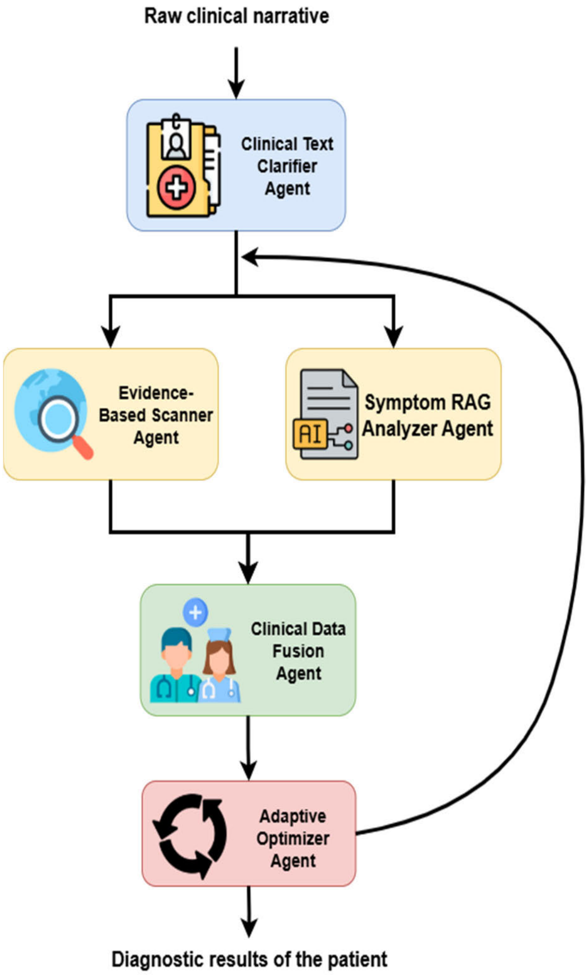

Raw clinical narrative
Clinical Text Clarifier Agent
Evidence-Based Scanner Agent
Symptom RAG Analyzer Agent
AI
Clinical Data Fusion Agent
Adaptive Optimizer Agent
Diagnostic results of the patient

FIGURE 2. Architecture of the proposed multi-agent CDSS.

During data processing, agents struggle to understand clinical input data in the form of plain user text. Therefore, the Clinical Text Clarifier tool sequentially processes the input text using a set of Natural Language Processing techniques to bring it into a structured form. The tool first uses Text Normalization to remove noise elements such as abbreviations (e.g., ''HT''), spelling errors, and punctuation from the raw data, and then converts the data into a standard text format. The standardized text is then passed to the Named Entity Recognition (NER) module. This module identifies and labels clinical entities within a set of predefined strings, such as ''Symptom,'' ''Disease,'' ''Medication,'' and ''Test Result.'' When simply recognizing entities is not sufficient, Association Extraction techniques are used to establish the semantic context between these entities. This allows a structured relationship to be established between the symptom ''Numbness'' and the anatomical region ''Left Arm,'' such as ''numbness in the left arm.'' In the final stage, the agent first maps all identified clinical concepts to codes in international medical ontologies such as SNOMED CT or ICD-10. At the end of this multi-step process, the agent converts the meaningless clinical text into a semantically enriched, structured format (e.g., JSON) that other agents in the system can directly process and that can be used for other LLM models.

The Symptom RAG Analyzer Agent receives structured clinical entities from the preprocessing step as input. It transforms these inputs into a semantic query vector, activating the RAG mechanism. It performs a cosine similarity-guided search on the vector database, identifying medical entities associated with the patient's clinical condition and generating a preliminary diagnosis list of corresponding possible diagnoses. In parallel, the Evidence-Based Scanner Agent acts to validate each hypothesis in this preliminary diagnosis list. This agent uses specialized tools to conduct systematic queries focused on academic databases such as PubMed and Google Scholar. Its goal is to refine the evidence base by finding the most current clinical guidelines, meta-analyses, and case reports for each possible diagnosis.

The Clinical Data Integration Agent acts as a synthesizer. It combines key diagnostic information from the vector database with current evidence from the literature review. It analyzes the information for consistency, flags conflicting findings, and calculates a confidence score for each diagnosis to produce a consolidated analysis report.

Finally, the Adaptive Optimization Agent is a quality check mechanism. If the reliability score of the generated report by the Integration Agent falls below a predetermined threshold ($\tau = 0.65$, where τ denotes the confidence threshold parameter), the agent creates a feedback loop. The feedback loop re-runs the process by paraphrasing the original query or increasing the search scope. This iterative process of optimization halts once the results meet the requirement of quality or the number of iterations has been exceeded. It keeps the system in the position to monitor itself and rectify it. That is, output quality is assured by a dynamic, feedback-based process of optimization, not by a static one-time test.

1) CLINICAL TEXT CLARIFIER AGENT

It serves a basic function in the first phase of the system. It accepts disorganized, panicked, or unstructured clinical data from patients and converts it into a standardized format. Along the way, it prioritizes the meaning of the original content while clarifying and highlighting critical diagnostic information by color-coding. It makes hastily written notes during emergencies, in particular, legible so that agents in downstream stages work with accurate information. This task is performed in a ''zero-shot'' setting, extracting relevant clinical entities from the input text and structuring them as a JSON object with a predefined schema. This agent directly helps the system succeed by maintaining data integrity. DeepSeek-R1 [26] was chosen for this task. The priority is error-free schema generation and entity/relationship normalization from freely written, cluttered clinical text. DeepSeek-R1 was chosen in our experiments because it yielded the highest schema concordance and the lowest error field rate (especially for ''Symptom-Anatomical Region'' relationships). Furthermore, its low latency prevents it from creating a bottleneck in the initial stage of the flow. The cost of this stage can be considered $\approx$O(L) for single pass normalization with the number of input text tokens L.

2) SYMPTOM RAG ANALYZER AGENT

This agent constructs diagnostic hypotheses based on a RAG architecture. The computation begins with input from the Clinical Text Clarifier Agent, receiving structured symptom data. The agent first converts the structured input into a highly semantically dense query vector. This is utilized for cosine similarity-based search against a custom vector database pre-indexed with medical textbooks, clinical guidelines, and epidemiological datasets. Thus, the most pertinent top-k query document passages are evidenced to be available in this search. Retrieving the top-k passages keeps the answer tied to material we actually index for the case at hand. Since approximate nearest neighbor search is used on Qdrant, the search time per query scales sublinearly with the collection size N, and linearly with the number of fetched passages k ($\approx$O(log N)+O(k)). Passages are ranked by cosine similarity between the query embedding and passage embeddings and only the top-k are passed as context. This reduces unsupported generalization and makes each proposed diagnosis traceable to a cited text span. In this paper, 'database' refers to the BioBERT–Qdrant vector index of textbook, guideline, and epidemiology passages, not a separate catalogue of diagnoses. The model suggests candidate conditions, which we retain only when at least one retrieved passage explicitly supports them.

This agent uses the Gemini-2.0-Flash [27] model because it generates evidence-based preliminary diagnostic rankings using chunks retrieved from a BioBERT-based vector database; the required capabilities are long-context semantic alignment and decision justification using RAG evidence. Gemini-2.0-Flash was selected because it provides the most balanced results in RAG context-consistent hypothesis generation and fast function call flows.

3) EVIDENCE-BASED SCANNER AGENT

This agent is designed to mitigate the risk of information in the system's static vector database becoming outdated over time and to leverage new methods. Rather than responding to a single patient query, the agent proactively keeps the system's knowledge base up-to-date by continuously scanning the medical literature.

To that end, the agent develops programmatic searches optimized for each hypothesis. It submits them via a web search API like Serper to scholarly indexes like PubMed and Google Scholar, and settled clinical guideline databases like NICE (National Institute for Health and Care Excellence). It derives text summaries pertinent to the purpose by interpreting web pages returned from the results obtained. One of the key roles is the evidence assessment process, whereby the information that has been gathered automatically gets categorized based on the hierarchy of evidence. In this process, higher-value evidence like systematic reviews and meta-analyses are weighted more than lower-value evidence like case reports.

The Qwen2-72B [28] model generates concise evidence summaries from dense texts and programmatic literature searches per hypothesis. When summarizing according to the weighted evidence hierarchy (guidelines/systematic reviews>case reports), Qwen2-72B was chosen for this task because it excels at both generating specific search queries and summarizing dense academic texts.

4) CLINICAL DATA FUSION AGENT

This is the core synthesis component of the system, and it produces the final analytical output. It takes two different data streams as input: (1) static, basic information from RAG and (2) dynamic, new evidence from the web crawler.

The agent's workflow begins by performing a cross-validation and consistency check between these two sources of data. For example, it regularly checks whether a treatment procedure in the textbook on which it is based still complies with the guidelines in the latest clinical guideline it has crawled by means of web crawling. It identifies inconsistencies in the data by executing a conflict detection protocol. It subsequently calculates a final calibrated confidence for each diagnostic hypothesis. It depends on factors such as the similarity score of the RAG, evidence score of web information, and degree of concordance between the two streams of information.

The key aspects of this task are reconciling conflicting evidence from RAG and the web, calibrating the confidence score, and generating an explainable report. Therefore, a structured, synthesized report summarizing the evidence, source, and final confidence level for each diagnosis adds explainability to the system. GPT-4o-mini [29] was selected for this agent because it provides a scalable synthesis of clinical data from various sources.

5) ADAPTIVE OPTIMIZER AGENT

This agent manages the workflow by monitoring the constantly changing nature of the diagnostic process. When confidence scores fall below a predetermined threshold, it identifies untrustworthy data and intelligently reorganizes query parameters, including keyword changes, query expansion/reduction, and other key points to investigate, generating a feedback report. It then uses this report to recommend more accurate queries to the Symptom RAG Analyzer and Evidence-Based Scanner agents. These agents, in turn, generate new queries based on this guidance. A feedback mechanism is used to optimize the data flow. This allows the system to quickly adapt to complex cases and increase its learning capacity. If the calculated confidence score exceeds a predetermined threshold, the loop is not entered.

In the feedback loop, instantaneous requery, query expansion/reduction, and parameter adaptation are necessary for low-confidence cases. Therefore, Gemini-2.0-Flash supports real-time iteration with low latency and high call throughput; it was chosen for this agent. The optimization loop halts when (i) the combined confidence score is above the predefined threshold ($\tau = 0.65$) or, (ii) the maximum of three iterations is achieved. In practice, these loops converge in 2–3 cycles, which had a mean response time of 3.7 minutes on the MedQA benchmark - we conclude this is practical to perform in routine, non-emergency clinical use.

The system proposed in this study is built to optimize clinical decision-making cycles on a multi-agent architecture organized in a modular organization. At its center is the harmony of RAG mechanism based on a static knowledge base and an evidence-scanning mechanism that continually updates said knowledge from the web. This two-tiered architecture repeatedly verifies the reliability of the outputs using a meta-controller agent-assisted iterative feedback mechanism. The whole process from unstructured data to evidence-based diagnostic synthesis is thus optimized for reliability and efficiency. The long-term objective is to maximize the quality of patient care through provision of clear and dynamic guidance to clinicians that not only provides an answer but also describes how the answer was derived.

B. BIOMEDICAL RAG ARCHITECTURE

This sub-section outlines the RAG architecture facilitating biomedical data integration and LLM agent interaction. The second key contribution of the study, the specialized RAG approach, is detailed in this section, which goes beyond general-purpose solutions and ensures the accurate interpretation of medical terminology and abbreviations. The developed architecture, based on biomedical domain-specific language models such as BioBERT, aims to maximize the semantic richness and accuracy of clinical texts. The architecture is composed of two parts: (1) a vector database from medical information sources and (2) expert tools that put together the database with LLM agents.

1) CREATING VECTOR DATABASE

A centerpiece of the proposed CDSS is a high-dimensional vector database comprising semantically enhanced medical knowledge. This database was built from diverse and reliable biomedical sources, particularly PubMed articles and symptom-disease datasets. This diversity allows the RAG mechanism to be understood not only in common but also in rare clinical conditions, reinforcing the model's inferential capabilities and domain-specific expertise. The framework can provide rapid as well as contextually consistent responses to natural language processing-based queries. As shown in Figure 3, the first step in the process is to painstakingly collect relevant sources of data; other steps involve dividing the texts for the sake of semantic coherence and eliminating undue content. Embeddings are then generated for each cleaned text sample, and finally a solid indexing strategy is employed to complete the vector database. Each phase is explained thoroughly in the following section.

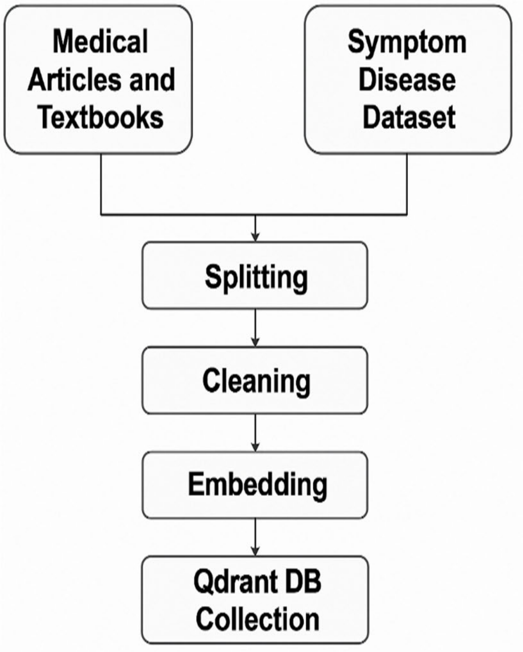


FIGURE 3. **Vector database creation pipeline.**

Questions loaded with over 15 keywords on issues of key importance such as correlation of symptoms and diagnosis, clinical findings patterns, and multi-organ injury were supplemented by filters that were biased towards human clinical studies and diagnostic papers. However, approximately 20 topics such as artificial intelligence applications, cancer biomarkers, and pandemic vaccine effects were excluded.

We downloaded up to 1,000 articles in parallel and ranked results based on a "relevance" ranking algorithm. Furthermore, Entrez's history option was activated in order to prevent overloading NCBI servers. The resulting publications were indexed locally in PDF format, and the parallelization approach supported this automation method with a 5.8x speedup over one CPU core in the pipeline, enabling the systematic compilation of clinical diagnostic literature.

To prepare each PDF file for text mining, PyMuPDF processed the text page by page at an average of 65 pages per second, ~9.6 times faster than pdfplumber, and extracted its content. During this stage, the text was corrected for typos and converted to a standard format. Because 28% of the pages required OCR, ICU-based Unicode normalization was performed. Reference lists, figure and table captions, copyright notices, author information, abstract and keyword sections, and publication-specific sections such as email addresses were automatically extracted using predefined regex and keyword filters.

The "Recursive Character Text Splitter" algorithm was used for text segmentation. The algorithm splits by setting "chunk_size = 512" and "chunk_overlap = 64" on the document. If in case the number of segments is inadequate, page-based fallback is activated, setting "chunk_size = 500" and "chunk_overlap = 80." This preserves both text fluidity as well as semantic coherence. The too brief and low-content segments and those with labels only are removed, and the remaining segments are tagged with metadata such as the source file name, the page number, and text length. Finally, every important chunk of text was translated to a vector utilizing the Sentence Transformer model BioBERT-mnli-snli-scinli-scitail-mednli-stsb. The vectors, together with the text information and pertinent metadata, were loaded into the Qdrant vector database engine as 768-dimensional vectors. It's worth noting that the creation and indexing of the vector db (document collection, embedding, and indexing) is all conducted offline as a one-off preprocessing task, so that during the online inference step, the system is only doing a pre-computed query on the vector index, providing an experience that is real-time responsive without having to repeat the heavy cost of executing the embeddings.

The Qdrant vector database enables fast response to clinical queries with its high-performance vector search, embedding storage, and semantic matching. A batch processing method was implemented in the loading stage; when the chunks were fewer, the content loss was minimized by employing small chunk sizes and other delimiters. Moreover, by enabling gRPC, query latency was reduced by 32% compared to REST. These processes produced 18M embeddings from 200,000 chunks for 1,000 PDFs. The final index size was 2.1 GB; total processing time was 38 minutes.

As limiting ourselves to text-based, information only may not provide us with sufficient semantic diversity within some disease categories, an open-label "symptom-disease" dataset [30] from the HuggingFace platform has been added to the system. It contains thousands of symptoms with corresponding relevant diagnoses and is thus extremely useful in primary care settings where clinical variability is the order of the day. Each data point is enriched with unique semantic patterns, leading to sentences that translate the diagnosis into different situations. The same diagnosis becomes more accessible in terms of different semantic representations in vector space.

This dataset was chosen because of the following:

1) As it is labeled, the system can generate more informative contextual embeddings.
2) It reflects the diversity of patients and symptoms in the real world.
3) This dataset contains some clinical conditions which are poorly represented in the literature

All points (vectors) of the database possess a unique ID and associated text. Moreover, as the system handles various sources of information, source and kind of text they represent (symptom-disease-dataset, medical-papers, etc. and diagnosis-symptom-pair, book-chunk, etc.) are stored as meta-information as well. That allows the system to grant source-related filtering and analysis capabilities. Figure 4 illustrates schematic conceptual decomposition of acquired symptom-diagnosis embeddings. The observed clusters on the graph are meant to portray graphically the model's ability to effectively pick up semantic similarities between symptoms and diagnoses.

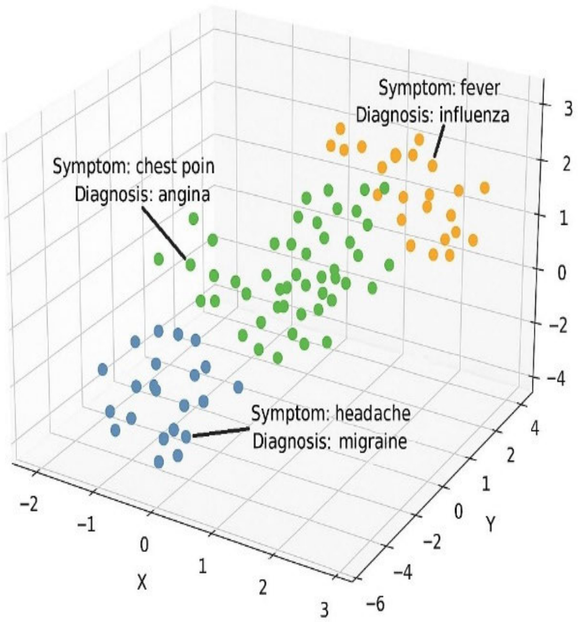


FIGURE 4. Schematic representation: Example of a grouping of symptom-diagnosis embeddings in 3D space. Blue dots represent the "headache-migraine" pair, green dots represent the "chest pain-angina" pair, and orange dots represent the "fever-influenza" pair.

2) CUSTOMIZED SEARCH TOOL FOR LLM AGENTS

Developed to improve the effectiveness of clinical decision support systems, the present work built a customized vector search tool that will enable LLM agents' fast and contextually harmonious access to medical knowledge. It integrates with the previously developed BioBERT-based Qdrant vector database and has the capability to supply dynamic responses to semantically rich medical queries. The tool is more sophisticated than basic vector matching methods and aims to achieve maximum query success using multilayered strategies.

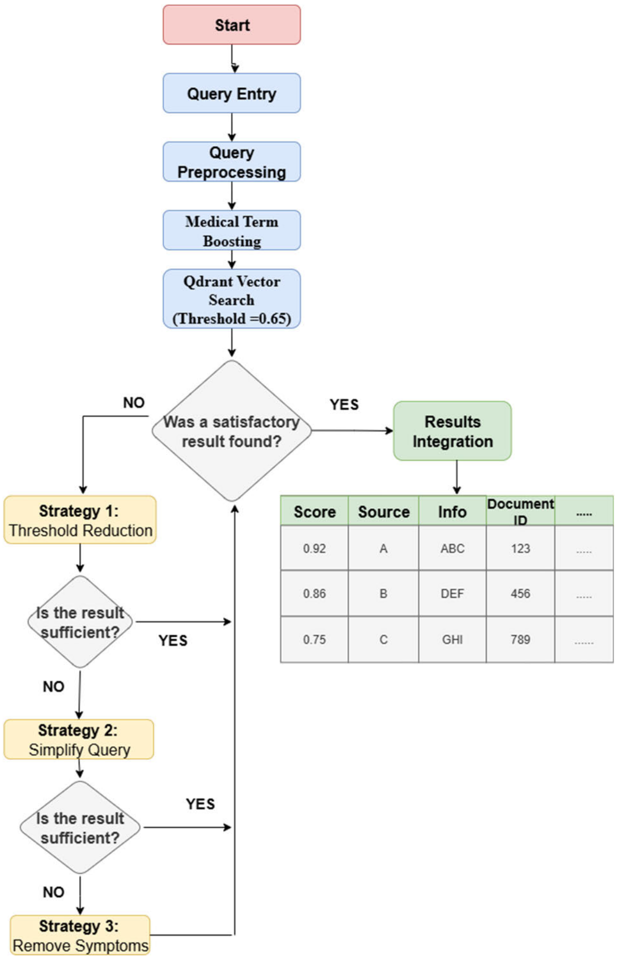
It calculates clinical term-specific embeddings from the BioBERT model to build a vector database and aligns these embeddings with different dimensions of semantic comparisons in Qdrant. Free-text natural language user input queries are normalized automatically, medical abbreviations are processed, and key terms are recognized which are preprocessed. The key innovation of one of the tools is its multi-level fallback strategies, which are triggered when the query does not succeed. At the first level, the similarity threshold is lowered to attempt larger matches. If there is still nothing, the query is linguistically simplified and re-embedded. Finally, the system re-constructs the query from the clinical semantic content of its store; i.e., it converts symptom patterns from the initial query and creates a new semantic cluster in order to create an alternative query. It is not constructed the same way as traditional "top-k" rotation methods, and this allows the system to be more flexible in the sense that it reveals learned behavior when handling inexact queries. The framework also employs a unique process known as "medical term boosting" to enhance content material of specific clinical terms. This facilitates repeated occurrence of high-priority symptoms in the query such that they gain more importance in embeddings. For example, such words of description like "shortness of breath" or "chest pain" are assigned higher scores in the vector space such that prioritized documents are marked.

The system is designed to be fully parametrically defined, directly invoked and customizable by agents, with respect to both the vector similarity threshold, maximum number of results, and context-based filtering options. Furthermore, to evaluate conflicting information from multi-source data, verifiable decision traces are created by maintaining detailed metadata related to queries (source, document ID, page, fragment length, retrieval time, etc.).

The technical flow of the developed tool is presented in Figure 5. As shown in the figure, the system first preprocesses the user query and activates medical term boosting, adjusted by the agent's decision. Then, embedding is generated and the vector database is searched. If the results obtained are insufficient or cannot be reproduced, automatic fallback strategies are activated, initiating a search for a more suitable match. When a sufficient result is found, the results contained in the metadata are transmitted to the agent. At the end of each query, the system returns results with high confidence scores to the agent.

C. CLINICAL VALIDATION AND INTEGRATION DATASETS, EVALUATION PROTOCOL AND BASELINES

Reliability and accuracy in clinical decision support systems can be achieved not only by the system generating information, but also by ensuring that this information is clinically valid, verifiable, and actionable. In the developed multi-agent architecture, this requirement is successfully met through


Start
Query Entry
Query Preprocessing
Medical Term Boosting
Qdrant Vector Search (Threshold =0.65)
NO
Was a satisfactory result found?
YES
Results Integration
Strategy 1: Threshold Reduction
Is the result sufficient?
YES
NO
Strategy 2: Simplify Query
YES
Is the result sufficient?
NO
Strategy 3: Remove Symptoms

Score	Source	Info	Document ID	.....
0.92	A	ABC	123	.....
0.86	B	DEF	456	.....
0.75	C	GHI	789	.....

FIGURE 5. **Adaptive multi-layered query processing pipeline for enhanced clinical decision support via BioBERT-Qdrant integrated semantic embeddings with fallback strategies.**

the tight integration of validation and integration processes within the modular agent structure.

The processes described under this heading directly address the third key contribution of the study: increasing the reliability of outputs by automatically combining and cross-checking data from WEB and RAG agents. This process, specifically managed by the Clinical Data Fusion Agent, ensures that the system provides evidence-based, consistent, and reliable recommendations.

The proposed system performs all operations, from the structuring of patient input to symptom analysis, from web-based data extraction to checking results for consistency, through a defined end-to-end clinical validation procedure. Three key agents perform this flow: the Symptom RAG Analyzer, the Evidence-Based Scanner, and the Clinical Data Fusion Agent. These agents, with specially designed Qdrant vector search utility and Serper web query module, produce an overall analysis from both literature and available medical resources. The parameters used during the analysis are summarized in Table 2. Figure 6 is a demonstration of a real patient scenario, marking the system operation with patient text, and the outputs produced by the agents step by step. This unstructured patient input is normalized first by the Clinical Text Clarifier Agent and then processed by the Symptom RAG Analyzer and the Evidence-Based Scanner. The Clinical Data Fusion Agent is employed to evaluate disagreement, whereas the Adaptive Optimizer constantly optimizes low-confidence findings.

TABLE 2. **Parameters used in the analysis.**

Component	Analysis Parameter	Value
RAG (retrieval)	Similarity criterion	Cosine Similarity
RAG (vector dimension)	Embedding dimension	768-D embedding
RAG (text segmentation)	chunk_size / overlap	512 / 64 (fallback: 500 / 80)
Term boosting	"Medical term boosting"	Active (baseline)
Adaptive feedback	Acceptance threshold	τ = 0.65
Adaptive feedback	Max. iterations	3

The system's data collection for the patient and its accuracy are guaranteed through the following methods:

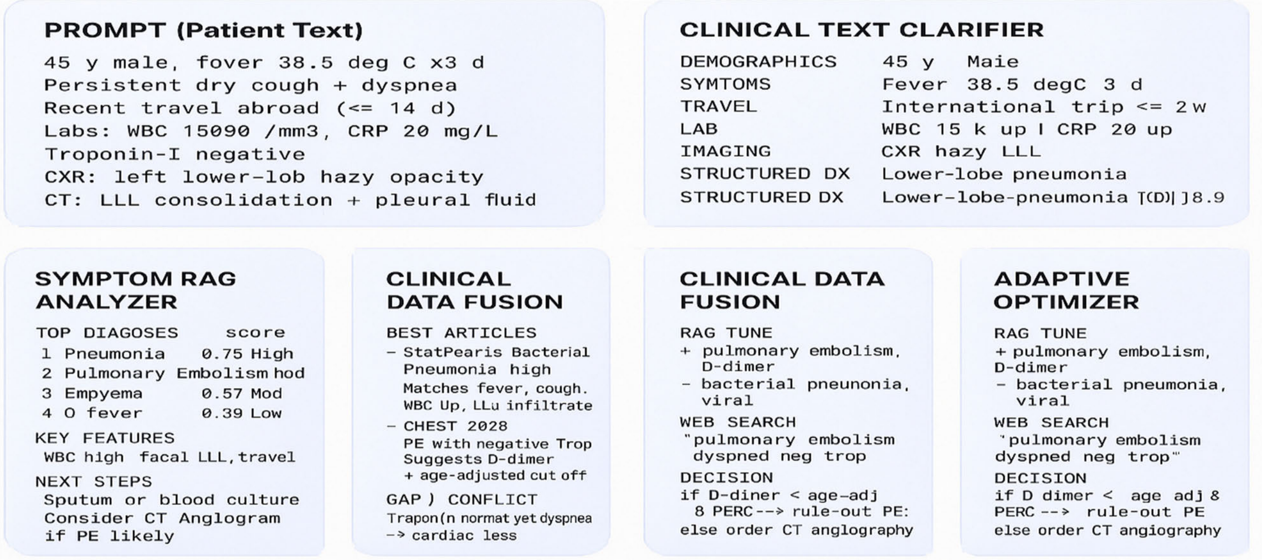
- Multi-Source Validation: Inferences based on the patient's symptoms are cross-validated with content from both vector database and academic web sources.
- Contextual Confidence Score: Each recommendation generated by the agents is evaluated by the Clinical Data Fusion agent using a contextual confidence score, and data below the threshold is selected and not made available to the agent.
- Semantic Fusion: The information obtained by the RAG and web agents is combined into a single aggregate, which we call the best data, through a consistency check.

On MedQA, the fused system reached 94.7%, outperforming RAG-only (65.0%) and web-only (70.0%) configurations by +29.7 and +24.7 points, respectively (Tables 5–6).

D. ADAPTIVE FEEDBACK LOOP

In dynamic clinical scenarios, the results of the initial analysis may not always be sufficiently reliable and accurate. In this case, one of the key innovations of the proposed system, the Adaptive Feedback Loop, comes into play, generating feedback reports that allow the system to learn from its own outputs and optimize for gaps in search queries or different scenarios that need to be investigated. This cyclical process is built on three main building blocks:

1) The Adaptive Optimizer Agent evaluates each analysis with a confidence score (e.g., $\tau = 0.65$, with τ representing the acceptance threshold for reliability). Results below the threshold are prompted for feedback due to the possibility of missing or incorrect information.



```
PROMPT (Patient Text)
45 y male, fover 38.5 deg C x3 d
Persistent dry cough + dyspnea
Recent travel abroad (<= 14 d)
Labs: WBC 15090 /mm3, CRP 20 mg/L
Troponin-I negative
CXR: left lower-lob hazy opacity
CT: LLL consolidation + pleural fluid
```

```
CLINICAL TEXT CLARIFIER
DEMOGRAPHICS    45 y  Maie
SYMTOMS         Fever 38.5 degC 3 d
TRAVEL          International trip <= 2 w
LAB             WBC 15 k up I CRP 20 up
IMAGING         CXR hazy LLL
STRUCTURED DX   Lower-lobe pneumonia
STRUCTURED DX   Lower-lobe-pneumonia ⁅(D)| ]8.9
```

```
SYMPTOM RAG
ANALYZER
TOP DIAGOSES        score
 1 Pneumonia       0.75 High
 2 Pulmonary Embolism hod
 3 Empyema         0.57 Mod
 4 O fever         0.39 Low
KEY FEATURES
 WBC high facal LLL, travel
NEXT STEPS
 Sputum or blood culture
 Consider CT Anglogram
 if PE likely
```

```
CLINICAL
DATA FUSION
BEST ARTICLES
– StatPearis Bacterial
  Pneumonia high
  Matches fever, cough.
  WBC Up, LLu infiltrate
– CHEST 2028
  PE with negative Trop
  Suggests D-dimer
  + age-adjusted cut off
GAP ) CONFLICT
Trapon(n normat yet dyspnea
-> cardiac less
```

```
CLINICAL DATA
FUSION
RAG TUNE
+ pulmonary embolism,
  D-dimer
– bacterial pneunonia,
  viral
WEB SEARCH
"pulmonary embolism
 dyspned neg trop
DECISION
if D-diner < age-adj
 8 PERC--> rule-out PE:
else order CT anglography
```

```
ADAPTIVE
OPTIMIZER
RAG TUNE
+ pulmonary embolism,
D-dimer
– bacterial pneumonia,
  viral
WEB SEARCH
'pulmonary embolism
dyspned neg trop"
DECISION
if D dimer <  age adj &
PERC -->  rule-out PE
else order CT angiography
```

FIGURE 6. This diagram displays the output of agents within the end-to-end functionality of the system for an example case. The Clinical Text Clarifier semantically annotates and structures unstructured patient data. The Symptom RAG Analyzer and Evidence-Based Scanner agents take the data and fetch diagnostic candidates from medical documents on the web and from trusted academic sources based on semantic vectors, respectively. Clinical Data Fusion agent analyzes data from those different sources and evaluates discrepancies and confidence levels. Adaptive Optimizer agent then re-directs the agents with rearranged queries, dynamically creating optimization recommendations for low confidence levels. This iterative process is intended to reinforce the system's diagnostic precision and reduce false-positive/false-negative rates. The figure demonstrates the integration of the modular and adaptable system framework into clinical decision support in general.

2) The iterative structure controlled by the feedback mechanism prompts the system to conduct re-investigations by providing new guidance reports to the agents up to a specified maximum number of iterations.
3) Feedback is not merely mechanical repetition; it consists of semantically structured recommendations generated by the contextual evaluation of the outputs.

The designed strategies enable the system to achieve the following features:

- Ability to automatically restructure queries that fail initial analysis.
- Ability to generate more appropriate alternative queries for ambiguous/missing symptoms.
- Exhibit reflexive behavior that optimizes itself by learning from case similarities over time.

Examining the case in Figure 6, the highly informative fused data from the Clinical Data Fusion Agent failed to meet the diagnostic confidence level in the initial evaluation by the Adaptive Feedback Loop Agent. To compensate for this deficit, some improvement suggestions were implemented for the RAG and Web Search Agents. To decrease ambiguities and enhance the variety of outputs, suggested query directions were intentionally embedded in the system. By adopting these suggestions, both the RAG and web agents began asking questions based on the suggestions. The number of iterations of the system was altered, and for the purposes of this study, it was set to a maximum of three cycles of iteration. Any more than three iterations would cost and take longer.

III. EXPERIMENTS & RESULTS

In this study, the proposed multimodal, multi-agent clinical decision support system was tested through comprehensive experimental analyses evaluated from different viewpoints. The study not only compared general accuracy rates, but also discussed the benefits of the system, including the diagnostic utility of its different components, the indispensable role of the adaptive learning process, and its potential to increase the reliability of clinical decisions.

All experiments were conducted on an HP Victus 16 laptop equipped with an Intel Core i5-14500HX processor (2.60 GHz, 14 cores), 16 GB RAM (5600 MT/s), and an NVIDIA GeForce RTX 4050 GPU with 8 GB VRAM. In the single case simulation, the total time increases approximately linearly with the number of fetched RAG passages k and the number of crawled web pages R, and the dominant part of the delay observed in practice is due to these two terms ($\approx O(k + R)$). The system had a 64-bit Windows 11 operating system with 954 GB SSD storage. All Python experiments were executed under the latest available versions of PyTorch, Transformers, and Qdrant libraries, ensuring reproducibility with up-to-date dependencies.

We implemented the proposed multi-agent model end-to-end on a prototype of an evidence-based web prototype. The prototype takes symptom/history input through a single-page interface and executes the agent structure we designed via REST APIs, generating an annotated report. For each run, the source utilized, potential sources of variation, and a calibrated confidence score would be automatically appended to the report; cases that fell below the confidence threshold prompting a feedback loop to reanalyze the case. This prototype operates in real-time and a typical case analysis takes 2–4 minutes; all interactions are retainable in auditable logs. To maintain data security, testing was performed with limited only synthetic, de-identified cases, no persistent patient data is retained, and access based on roles were restricted. This applied deliverable demonstrates that the model is not just theoretical design; it also works with a true end-to-end workflow.

To evaluate the system's essential performance in an objective way, three different biomedical natural language understanding datasets, well-liked within the literature: MedQA, PubMedQA, and MedBullets, were selected.

MedQA is a dataset consisting of multiple-choice questions covering postgraduate medical knowledge. PubMedQA is a more analytical dataset derived from biomedical articles, requiring answers to research questions and abstracts. MedBullets offers an evaluation environment based on practical knowledge, primarily based on basic medical sciences and case studies.

When these three datasets are used together, an accurate view of the system's performance across different types of knowledge and levels of reasoning can be obtained. Because these datasets are used for comparative purposes in studies, they also allow for objective comparison of the proposed framework with other solutions in the literature. Table 3 summarizes brief technical specifications of the datasets used and the number of samples evaluated in each.

TABLE 3. **Dataset summaries and number of tests.**

Dataset	Short technical specifications	Number of tests
MedQA	Postgraduate medical knowledge; multiple choice (A–E); diagnostic reasoning measure	n=50
PubMedQA	Derived from biomedical articles; inference from research question + abstract; free-text answer	n=50
MedBullets	Applied/basic medicine and case-based short questions; free-text response	n=50

According to the results presented in Table 4, while traditional LLM-based systems without an agent architecture remained at average accuracy, single-model agent systems achieved a certain improvement. In this study, we took care to keep the number of data used in the evaluations equal to ensure objective comparison of the performance of the developed system with the accuracy levels of structures previously tested for similar studies in the literature. Therefore, while some studies in the iteration (no agent category) presented evaluation results on different numbers of samples in the MedQA, PubMedQA, and MedBullets datasets, systems such as MDAgents, Reconcile, and AutoGen (categories other than non-agent) were all tested on 50 samples [31].

To give experimental validity to the proposed system and to enable equitable comparison with existing research work in the literature, we evaluated 50 questions per dataset on MedQA, PubMedQA, and MedBullets under the same settings. This allows the results to be compared fairly against similar studies.

We present micro-averaged values of precision, recall (sensitivity), F1-score, and specificity in addition to accuracy based on total counts of true positives (TP), false positives (FP), true negatives (TN), and false negatives (FN). In the 5-choice multiple-choice MedQA task, we used a per-question and one-vs-all reduction: the one correct option is treated as the positive class and the four remaining options are treated as negatives. A correct prediction contributes TP=1 and TN=4 contributions, and an incorrect prediction contributes FP=1, FN=1, and TN=3 contributions. Everything is summed over all questions; in this formulation, micro-precision, micro-recall, and micro-F1 are equal to accuracy by construction, and specificity quantifies the power to reject the four non-gold options.

For datasets with free-text responses (PubMedQA and MedBullets), predictions are mapped to a binary correctness decision via a fixed semantic-similarity rule. Let (·) be the sentence-embedding function, $u \;\; = (\hat{\boldsymbol{y}})$ for the predicted answer and $v = (y)$ for the reference. The cosine similarity is $\mathrm{s}(\hat{\boldsymbol{y}}, \mathrm{y}) = \cos(u, v) = \cos(\phi(\hat{\boldsymbol{y}}), \phi(\mathrm{y}))$=($u \cdot v$)/(‖ u ‖ · ‖ v ‖), where $u = \phi(\hat{\boldsymbol{y}})$ and $v = \phi\ (\ y\ )$ so $s \in [\,-\,1,\,1\,]$ s∈[−1,1]. We fix $\theta\ = 0.80$ a priori to reflect high semantic agreement and to balance precision–recall on a held-out development split: lower values inflated false positives, while higher values substantially reduced recall. Under this binary criterion, micro-averaged precision, recall, and F1-score coincide with accuracy; we nevertheless report them explicitly for completeness (see Table 5).

We used the dynamic architecture that we developed on three varied datasets with up to three iterations. In this study, accuracy refers to exact-match accuracy for the multiple-choice dataset (MedQA) and semantic-similarity-based correctness for free-text datasets (PubMedQA, MedBullets). Our experiment yielded accuracy scores that were better than previous works in the field: 94% on MedQA, 88% on PubMedQA, and 84% on MedBullets. The accuracy score indicates that the system optimizes information retrieval as well as contextual adaptation successfully. The experiments indicate performance gains over prior work across all three datasets. We attribute the improvement to (i) domain-specific retrieval, (ii) the fusion controller that cross-checks RAG and web evidence, and (iii) the feedback loop that re-queries low-confidence cases.

Three key reasons stand out in achieving the desired performance level. First, generating qualified queries based on vectors containing multi-layered semantic relationships is crucial. Second, the effective optimization approach used to combine these queries and the agent algorithms supporting incremental learning improve accuracy and efficiency. Finally, a rational division of labor among agents and the selection of the most appropriate large language model for each task deepened the expertise of each agent,

TABLE 4. Comparison of models on MedQA, PubMedQA and MedBullets datasets.

Category	Model Name	Datasets			Related Studies
		MedQA	PubMedQA	MedBullets	
No agent	Llama 3	78.1	66.3	68.5	[16], [31],
	GPT-4	79.6	75.4	66.2	[31], [32]
	OpenBioLLM-70B	75.1	79.3	58.4	[33]
	DeepSeek-R1	92.0	76.2	79.2	[34]
	QwQ-32B	78.6	77.8	54.2	[34]
Adaptive	MDAgents	88.7	75.0	80.8	[18]
Multi-agent (Single-model)	ER	81.9	56.0	76.0	[18]
	Medprompt	82.4	51.8	71.0	[18]
Multi-agent (Multi-model)	Reconcile	81.3	79.7	59.5	[18]
	AutoGen	60.6	77.3	55.3	[18]
	DyLAN	64.2	73.6	57.3	[18]
	This Study	**94.0**	**88.0**	**84.0**	This

TABLE 5. Overall and micro-averaged metrics.

Dataset	MedQA	PubMedQA	MedBullets
N (evaluated)	50	50	50
Correct	47	44	42
Incorrect	3	6	8
Accuracy	0.94	0.88	0.84
Precision (micro)	0.94	0.88	0.84
Recall (micro)	0.94	0.88	0.84
F1-score (micro)	0.94	0.88	0.84
Specificity	0.985	0.88	0.84

improving output quality by 14.2%. The proposed adaptive optimization paradigm automatically reprocessed outputs below a confidence threshold of 0.65 to improve the queries of other agents, thus increasing system consistency by 1.8 times.

Table 6 shows 20 classically prepared questions from MedQA, posed to the system using different iteration numbers, and their accuracy values compared. For MedQA we report exact-match accuracy on the multiple-choice label (A–E). For PubMedQA and MedBullets, which yield free-text answers, we report semantic similarity; we also include exact-match where applicable. This method was chosen because it assesses the equivalence of texts at the content and context levels using vector space using metrics such as cosine similarity. The correct and generated answers to the classical questions posed to the system were analyzed, and the average results are presented in Table 6.

As shown in Table 6, accuracy rises from 86.3% at the first pass to 94.7% by the third, while the marginal gain beyond the third pass is small. This pattern is consistent with the feedback loop claim: the system improves with use, but the improvement in accuracy slowed down with an increase in the number of iterations, and thus the processing times were increased logarithmically. Minimal changes were observed in the fourth and fifth cycles, up to the optimal level of performance. The reason the feedback loop begins to repeat the same suggestions is the main reason for the lack of a significant increase in accuracy with further iterations.

TABLE 6. System performance at different iteration numbers.

Iteration	Accuracy	Processing Time
1	86.3% (±1.1)	2.5 minutes
2	91.5% (±0.8)	3.0 minutes
3	94.7% (±0.5)	3.7 minutes
4	95.1% (±0.4)	4.4 minutes
5	95.5% (±0.3)	5.5 minutes

In our experiments, we measured the time consumed approximately as follows: for a single query cycle (RAG call + web crawl + fusion), the request triggered by the RAG agent took ∼1 s, the response from the vector lookup took ∼1 s, and the agent processed and decided on this response took ∼10 s on average; output transfer between agents was negligible (<1 s). The web agent added an average delay of ∼45 s for a single page; because most questions required 1–2 additional pages to be visited depending on the page content (and the RAG agent could regenerate queries based on the findings), this portion could extend to a total of ∼1–2 minutes. Together, these components increased the typical wall-clock time for a single iteration to ∼2–3 minutes, and to ∼3–4 minutes when 2–3 iterations were applied; the total time for the 50-question evaluation (including reviewing and annotating the results) was ∼2–3 hours in practice. The time variance was primarily due to the web agent's multiple-source crawling and requeries triggered at low confidence scores; the evaluation time, depending on model differences, was on average in the range of ∼8–12 s. These measurements show that the system's temporal cost is largely due to web access and iterative optimization steps, while inter-agent communication does not contribute significantly to the total delay.

To investigate the contribution of performance of each agent within the system, an ablation study was conducted using 20 test questions from a MedQA dataset. One agent at a time was removed from the system, and the remaining part of the database was queried with the same 20 questions. Table 7 results show significant effects on system performance caused by the Adaptive Optimizer and Clinical Data Fusion agents. For example, when the Evidence-Based Scanner Agent was turned off, the system generated more stale and out-of-scope recommendations. Such an analysis clearly demonstrates the complementary and synergistic role of the modular design in the suggested framework.

TABLE 7. Change in system performance when agent components are turned off.

Component removed	Accuracy	Deficiency in the system
Clinical Text Clarifier	80.0%	Semantic distortion in input texts made symptom detection difficult.
Symptom RAG Analyzer	65.0%	The production of literature-based diagnoses has weakened significantly.
Evidence-Based Scanner	70.0%	Current clinical information could not be obtained from the web, so old information was used.
Clinical Data Fusion	60.0%	Conflicting data could not be combined, and inconsistency in recommendations was observed.
Adaptive Optimizer	70.0%	Low confidence answers were given without correction, and the system's learning reflex was lost.
Medical Term Boosting(Disabled)	80.0%	Prioritization of critical clinical expressions, diagnostic accuracy decreased.

The information acquired shows that the system is not composed of independently operative pieces, but rather becomes a unified whole because of the competence developed by each agent in its specific area of responsibility.

For example, deactivating the Adaptive Optimizer component shuts down the system's ability to scan and optimize its performance over time, and as a result, the precision in the decisions being made was severely lessened. Similarly, deactivating the Clinical Data Fusion module from the system impaired the convergence of knowledge from various sources, which resulted in either varied or downright wrong outputs.

The experimental results strongly confirm the three main contributions of the study. First, the dynamic self-optimization capability of the proposed flexible, multi-agent architecture is demonstrated by the increase in accuracy from 86.3% to 94.7% after three iterations on the MedQA dataset (Table 6). Furthermore, the accuracy dropped to 70.0% after removing the "Adaptive Optimizer" agent in the ablation study, demonstrating that this agent is critical to the system's performance (Table 7).

Secondly, the value of the customized BioBERT-based RAG approach in correctly interpreting medical terminology was confirmed by the ablation studies. Removing the "Symptom RAG Analyzer" agent, which forms the basis of the RAG mechanism, reduced accuracy to 65.0%, while disabling the "Medical Term Strengthening" feature reduced it to 80.0%, highlighting the importance of this customized approach.

Finally, the ability to increase output reliability by combining data from WEB and RAG agents was supported by an ablation study. Removing the 'Clinical Data Fusion' agent responsible for data fusion resulted in inconsistent recommendations, reducing accuracy to 60.0% and causing the most significant performance degradation. This finding demonstrates that cross-checking and data fusion are essential for reliability. Altogether, these results confirm that multi-source data collection and fusion are essential for high accuracy and reliability.

All these conclusions demonstrate that the utilized multi-agent LLM architecture, besides providing technical competence, also provides significant practical benefits. The system not just enhances accuracy for clinical decision support tasks, but decision reliability and contextual appropriateness as well. Additionally, since the system is modular, it can be readily modified to different circumstances, whereas its open-to-learning-based execution gives a more adaptative platform than existing solutions.

IV. LIMITATIONS AND DISCUSSION

The study showed that a multi-agent based LLM architecture can enhance accuracy and trust in clinical decision support systems. The method presented provides drastic improvements compared to similar approaches in the literature. Overall, the methodological scope of the study and applicability restrictions are to be considered.

Evaluations were performed first on three generic datasets (MedQA, PubMedQA, and MedBullets). Although all datasets consist of different types of clinical information, none of them widely cover the range of real-world patients. For that reason, later clinical studies should investigate how the system performs with rare cases or multi-disease complexity.

Furthermore, while it is a benefit that the platform's architecture can use web and literature searches to provide up-to-date information, the level of effectiveness will depend on the quality and accessibility of the actual information sources used. Future research will limit this limitation with hybrid solutions capable of integrating with official clinical guidelines and evaluating information source reliability.

Furthermore, while the multi-agent architecture increases the system's flexibility, its computational cost is higher than that of single models. This can be limiting, especially in time-critical clinical environments. This limitation can be overcome by using more efficient models or cloud-based integrations.

Lastly, we acknowledge the fact that the system proposed is primarily a tool to support clinical decision-making, and not a standalone diagnostic tool; the conclusion for the course of action lies ultimately with the expert physician. Although the risk of hallucination in LLMs cannot be removed entirely, by employing data fusion and a feedback loop developed in this work to focus on the hallucination risks, we were able to minimize the risk significantly.

Overall, the findings demonstrate that this architecture can provide reliable, up-to-date, and interpretable support to clinicians. Future studies should focus on multicenter clinical validation, resource-efficient optimizations, and strengthening the ethical/regulatory dimension.

V. CONCLUSION

This study proposes a new architecture consisting of multiple large language model agents to make clinical decision support systems more flexible, precise, and contextually responsive. Unlike rigid solutions in the literature, the designed system involves semantic analysis, contextual inference, and adaptive feedback loops to react to dynamic clinical conditions.

The quantitative aspects of this architecture clearly demonstrate success in that the accuracy targets were achieved (reporting 94% accuracy on MedQA, 88% on PubMedQA, and 84% on MedBullets) all of which represent a substantial improvement from the previously defined benchmarks. We attributed this performance gain to the design principles underpinning the system namely, the combined access to data from a specialized biomedical RAG to enrich dynamic web searches, and the continuous adaptation provided the feedback loop which increased our accuracy from 86.3% to 94.7% on the MedQA dataset. This research has promising implications for the practice of clinical medicine. A multi-source validated, synthesized analysis is a real opportunity to mitigate the diagnostic errors associated with cognitive overload. The architecture can serve as a form of "clinical intelligence partner", which builds upon a physician's cognitive schema by quickly and effectively synthesizing a multi-faceted complex patient presentation, bridging symptoms to the evidence base, while establishing connection to both the scope of decision making by ensuring provision of an overall contextual framework of knowledge that is always current. This capacity may be especially relevant in high-stakes clinical encounters. Taking on a role in the diagnostic process will improve not only diagnostic accuracy, but also ultimately, patient safety and clinical outcomes.

Empirical evaluations demonstrate the system to have statistically significant performance improvement over state-of-the-art methods on MedQA, PubMedQA, and MedBullets datasets. Performance metrics demonstrate superiority in transparency, consistency, and learnability of decision-making processes, in addition to improved accuracy. Role optimization within the agent-based architecture enhances system-level synergy through the assignment of singular tasks to the area of expertise of the respective component. Ablation studies revealed that modular agent interactions improve decision quality. The adaptive feedback mechanism, by automatic reconfiguration of low-confidence cases, improves error tolerance and thus enables reliability in clinical practice. These findings demonstrate that LLMs can be used as dynamic support systems that can be combined with clinical expertise, and not as just data processing tools.

Limitations of the study are extra computational load in real-time clinical environments and repeated process resource utilization. Further, variability of model performance in rare clinical environments and hallucinations risk should be questioned. Multicenter clinical testing and expert interactive assessment protocols are to be evaluated with subsequent research to test the system generalizability. The proposed design is envisioned to provide a framework for constructing ethical and sustainable decision support systems for health practitioners.

ACKNOWLEDGMENT

This study was conducted within the scope of Ph.D. thesis of Kübra Arslanoğlu.

REFERENCES

[1] M. A. Musen, B. Middleton, and R. A. Greenes, "Clinical decision-support systems," in *Biomedical Informatics: Computer Applications in Health Care and Biomedicine*. Cham, Switzerland: Springer, 2021.

[2] M. J. Rahim, A. Afroz, and O. Akinola, "Predictive analytics in healthcare: Big data, better decisions," *Int. J. Sci. Res. Mod. Technol.*, vol. 4, pp. 1–21, Apr. 2025.

[3] Y. Ohta, I. Miki, T. Kimura, M. Abe, M. Sakuma, K. Koike, and T. Morimoto, "Epidemiology of adverse events and medical errors in the care of cardiology patients," *J. Patient Saf.*, vol. 15, no. 3, pp. 251–256, Sep. 2019.

[4] R. Greenes, *Clinical Decision Support: The Road to Broad Adoption*. Cambridge, U.K.: Cambridge Univ. Press, 2014.

[5] J. Achiam et al., "GPT-4 technical report," 2023, *arXiv:2303.08774*.

[6] C. U. Ogdu, M. Yilmazer, and M. Karakose, "A deep learning-based approach for damage detection in cultural heritage images and generating heat maps," in *Proc. 2nd Int. Conf. Sustaining Heritage, Embracing Technol. Advancements (ICSH)*, Sep. 2024, pp. 1–5.

[7] A. Matarazzo and R. Torlone, "A survey on large language models with some insights on their capabilities and limitations," 2025, *arXiv:2501.04040*.

[8] K. Arslanoğlu and M. Karaköse, "A trustworthy analysis approach for chatbots on health data: ChatGPT-4 example," in *Proc. 29th Int. Conf. Inf. Technol. (IT)*, Zabljak, Montenegro, Feb. 2025, pp. 1–4, doi: 10.1109/it64745.2025.10930304.

[9] A. Nayak, M. S. Alkaitis, K. Nayak, M. Nikolov, K. P. Weinfurt, and K. Schulman, "Comparison of history of present illness summaries generated by a chatbot and senior internal medicine residents," *JAMA Internal Med.*, vol. 183, no. 9, pp. 1026–1027, Sep. 2023.

[10] J. W. Ayers, A. Poliak, M. Dredze, E. C. Leas, Z. Zhu, J. B. Kelley, D. J. Faix, A. M. Goodman, C. A. Longhurst, M. Hogarth, and D. M. Smith, "Comparing physician and artificial intelligence chatbot responses to patient questions posted to a public social media forum," *JAMA Internal Med.*, vol. 183, no. 6, pp. 589–596, Jun. 2023.

[11] M. Karaköse, H. Yetiş, and M. Çeçen, "A new approach for effective medical deepfake detection in medical images," *IEEE Access*, vol. 12, pp. 52205–52214, 2024, doi: 10.1109/ACCESS.2024.3386644.

[12] P. C. Sherimon and R. Krishnan, "OntoDiabetic: An ontology-based clinical decision support system for diabetic patients," *Arabian J. Sci. Eng.*, vol. 41, no. 3, pp. 1145–1160, Mar. 2016.

[13] Z. Ji, D. Chen, E. Ishii, S. Cahyawijaya, Y. Bang, B. Wilie, and P. Fung, "LLM internal states reveal hallucination risk faced with a query," 2024, *arXiv:2407.03282*.

[14] A. G. Levra, M. Gatti, R. Mene, D. Shiffer, G. Costantino, M. Solbiati, R. Furlan, and F. Dipaola, "A large language model-based clinical decision support system for syncope recognition in the emergency department: A framework for clinical workflow integration," *Eur. J. Internal Med.*, vol. 131, pp. 113–120, Jan. 2025.

[15] A. Sokolov, "Real-time data analytics in medical device software: Enhancing clinical decision support systems," *Sci. Academia J.*, vol. 4, pp. 1–10, Apr. 2021.

[16] C. U. Ogdu, S. Gurbuz, M. Karakose, and E. Hanoglu, "Medical implications of LLM based clinical decision support systems in healthcare," in *Proc. 29th Int. Conf. Inf. Technol. (IT)*, Feb. 2025, pp. 1–4.

[17] K. Zuo, Y. Jiang, F. Mo, and P. Liò, "Kg4diagnosis: A hierarchical multi-agent LLM framework with knowledge graph enhancement for medical diagnosis," in *Proc. AAI Bridge Program AI Med. Healthcare*, vol. 281, 2025, pp. 195–204.

[18] Y. Kim, C. Park, H. Jeong, Y. S. Chan, X. Xu, D. McDuff, H. Lee, M. Ghassemi, C. Breazeal, and H. W. Park, "Mdagents: An adaptive collaboration of LLMS for medical decision-making," in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 37, 2024, pp. 79410–79452.

[19] N. Mehandru, B. Y. Miao, E. R. Almaraz, M. Sushil, A. J. Butte, and A. Alaa, "Large language models as agents in the clinic," 2023, *arXiv:2309.10895*.

[20] S. Wilk, M. Kezadri-Hamiaz, D. Rosu, C. Kuziemsky, W. Michalowski, D. Amyot, and M. Carrier, "Using semantic components to represent dynamics of an interdisciplinary healthcare team in a multi-agent decision support system," *J. Med. Syst.*, vol. 40, no. 2, pp. 1–12, Feb. 2016.

[21] J. C. L. Ong, L. Jin, K. Elangovan, G. Y. S. Lim, D. Y. Z. Lim, G. G. Ren Sng, Y. Ke, J. Y. M. Tung, R. J. Zhong, C. M. Y. Koh, K. Z. H. Lee, X. Chen, J. K. Chng, A. Than, K. J. Goh, and D. S. W. Ting, "Development and testing of a novel large language model-based clinical decision support systems for medication safety in 12 clinical specialties," 2024, *arXiv:2402.01741*.

[22] J. Chen, H. Lin, X. Han, and L. Sun, "Benchmarking large language models in retrieval-augmented generation," in *Proc. AAAI Conf. Artif. Intell.*, vol. 38, no. 16, Mar. 2024, pp. 17754–17762.

[23] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So, and J. Kang, "BioBERT: A pre-trained biomedical language representation model for biomedical text mining," *Bioinformatics*, vol. 36, no. 4, pp. 1234–1240, Feb. 2020.

[24] A. J. McMurry, D. Phelan, B. E. Dixon, A. Geva, D. Gottlieb, J. R. Jones, M. Terry, D. Taylor, H. G. Callaway, S. Mahoharan, T. Miller, and K. D. Mandl, "Large language model symptom identification from clinical text: A multi-center study," *MedRxiv*, Dec. 2024, doi: 10.1101/2024.12.16.24319044.

[25] D. M. Levine, R. Tuwani, B. Kompa, A. Varma, S. G. Finlayson, A. Mehrotra, and A. Beam, "The diagnostic and triage accuracy of the GPT-3 artificial intelligence model: An observational study," *Lancet Digit. Health*, vol. 6, no. 8, pp. e555–e561, Aug. 2024.

[26] D. Guo et al., "DeepSeek-r1: Incentivizing reasoning capability in LLMs via reinforcement learning," 2025, *arXiv:2501.12948*.

[27] R. Balestri, "Gender and content bias in large language models: A case study on Google Gemini 2.0 flash experimental," *Frontiers Artif. Intell.*, vol. 8, Mar. 2025, Art. no. 1558696.

[28] X. Tian, S. Zhao, H. Wang, S. Chen, Y. Ji, Y. Peng, H. Zhao, and X. Li, "Think twice: Enhancing LLM reasoning by scaling multi-round test-time thinking," 2025, *arXiv:2503.19855*.

[29] M. N. A. Siddiky, M. E. Rahman, M. F. B. Hossen, M. R. Rahman, and M. S. Jaman, "Optimizing AI language models: A study of ChatGPT-4 vs. ChatGPT-4o," Preprint, Feb. 2025, doi: 10.20944/preprints202502.0066.v1.

[30] D. Tecblic. *Symptom-Disease Dataset*. Hugging Face. Accessed: Aug. 22, 2023. [Online]. Available: https://huggingface.co/datasets/duxtecblic/symptom-disease-dataset

[31] H. Chen, Z. Fang, Y. Singla, and M. Dredze, "Benchmarking large language models on answering and explaining challenging medical questions," in *Proc. Conf. Nations Americas Chapter Assoc. Comput. Linguistics, Hum. Lang. Technol.*, Apr. 2025, pp. 3563–3599.

[32] H. Feng, F. Ronzano, J. LaFleur, M. Garber, R. de Oliveira, K. Rough, K. Roth, J. Nanavati, K. Zine El Abidine, and C. Mack, "Evaluation of large language model performance on the biomedical language understanding and reasoning benchmark: Comparative study," *MedRxiv*, 2024, doi: 10.1101/2024.05.17.24307411.

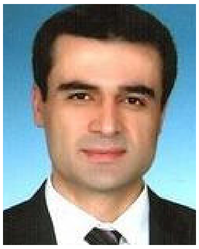
[33] X. Huang, J. Wu, H. Liu, X. Tang, and Y. Zhou, "M1: Unleash the potential of test-time scaling for medical reasoning with large language models," 2025, *arXiv:2504.00869*.

[34] X. Tang, D. Shao, J. Sohn, J. Chen, J. Zhang, J. Xiang, F. Wu, Y. Zhao, C. Wu, W. Shi, A. Cohan, and M. Gerstein, "MedAgentsBench: Benchmarking thinking models and agent frameworks for complex medical reasoning," 2025, *arXiv:2503.07459*.

[35] Y. Feng, "Semantic textual similarity analysis of clinical text in the era of LLM," in *Proc. IEEE Conf. Artif. Intell. (CAI)*, Singapore, Jun. 2024, pp. 1284–1289.

ÇAĞATAY UMUT ÖĞDÜ is currently pursuing the master's degree in artificial intelligence and quantum algorithms. He is a Computer Engineer at the Faculty of Engineering, Fırat University, Elâzığ, Türkiye. He graduated top of both his faculty and department in 2025. His research interests include smart cities, artificial intelligence, data science, image processing, large language models, and quantum technologies. He actively participates in local competitions on sustainable and AI technology solutions and has achieved various awards.

KÜBRA ARSLANOĞLU (Member, IEEE) received the B.S. degree in statistics from Giresun University, in 2012, and the B.S. degree in computer engineering and the M.S. degree in biostatistics from Fırat University, in 2020. Since 2020, she has been a Research Assistant with the Department of Software Engineering, Fırat University. Her research interests include blockchain, trustworthy artificial intelligence, and artificial intelligence applications in healthcare systems.



MEHMET KARAKÖSE (Senior Member, IEEE) received the B.S. degree in electrical engineering and the M.S. and Ph.D. degrees in computer engineering from Fırat University, Elâzığ, Türkiye, in 1998, 2001, and 2005, respectively. From 1999 to 2005, he was a Research Assistant with the Department of Computer Engineering, Fırat University. He was an Assistant Professor and an Associate Professor with Fırat University, from 2005 to 2014 and from 2014 to 2020, respectively. He is currently a Professor Doctor with the Department of Computer Engineering, Fırat University. His research interests include fuzzy systems, intelligent systems, quantum computing, simulation and modeling, fault diagnosis, computer vision, railway inspection systems, and photovoltaic systems.