MediLink: A Multi-Agent Conversational Pipeline for Evidence-Grounded Symptom Diagnosis

Master's Project Report

Submitted by:

Bhushan Mahajan
Master's in Computer Science
University at Buffalo, State University of New York

Project Advisor:

Professor Kaiyi Ji
University at Buffalo, State University of New York

May 12, 2025

Abstract

MediLink is a multi-agent conversational chatbot developed to provide preliminary medical diagnosis support through interactive symptom gathering and evidence-based reasoning, without aiming to replace human medical professionals. Unlike traditional symptom checkers or basic symptom-to-disease mapping systems, MediLink actively engages users in iterative dialogue, dynamically refining its diagnostic predictions with each interaction and in a more nuanced diagnosis. It operates in two distinct modes: a dedicated clinical Q&A mode for symptom assessment and diagnosis, and a casual chat mode for general conversations with the user. Its core diagnostic engine combines semantic search and AI-generated predictive models, with the generative approach (using MedAlpaca-7B with QLora) achieving approximately 95.4% prediction accuracy during internal evaluations while training and testing. To ensure more robust predictions, MediLink uses Retrieval Aggregated Generation (RAG), encoding user-reported symptoms as vectors to retrieve accurate disease information from trusted medical knowledge bases instead of relying solely on the internal knowledge of a language model. As users respond, the system iteratively updates disease confidence scores and employs a symptom co-occurrence matrix to identify critical follow-up questions, systematically reducing diagnostic ambiguity. Once MediLink's confidence exceeds a predefined threshold that is 90%, the chatbot shifts into a comprehensive evidence-based reasoning phase. Here, it uses not only the demographics of the patient, which we collect at the start of the session, but also the cumulative set of symptoms extracted throughout the conversation with the user, as well as the most likely predicted disease, to obtain relevant and up-to-date medical literature from PubMed. By segmenting these recovered articles into manageable chunks, MediLink performs chain-of-thought (CoT) reasoning, explicitly explaining its diagnosis, suggesting precautions, and transparently detailing its rationale. Recognizing that current online symptom searches typically force users to sift through numerous disconnected articles, often missing critical context or detail, MediLink provides structured, coherent preliminary guidance. Although its predictions might be constrained by the limitations of the underlying knowledge graph, the system offers users a valuable starting point, laying a foundation for further clinical evaluation and potentially serving as a stepping-stone in a broader diagnostic chain. Future enhancements, such as integrating symptom severity (mild, high, acute, chronic) and patient medical history, could substantially boost accuracy, although current datasets lack such detailed temporal and historical dimensions. Ultimately, MediLink exemplifies how intelligent conversational AI can surpass generic search-based queries or simplistic chatbot responses, paving the way for more personalized, context-aware preliminary medical assessments.

Contents

Introduction

Artificial intelligence is increasingly being used to assist in medical diagnosis, leveraging large language models (LLMs) for their strong reasoning and conversational abilities. However, purely generative LLM-based approaches can suffer from issues such as hallucinations and a lack of interpretability and transparency in decision making [10]. Ensuring precision and clarity in a user-facing diagnostic system is critical, as clinical decisions demand correct outcomes and understandable justifications. Traditional symptom-checker systems rely on predefined decision trees or probabilistic models, which may lack flexibility and conversational ability. Newer approaches explore LLMs for differential diagnosis, but integrating them effectively into an interactive tool remains challenging.

MediLink is proposed as a multi-agent conversational diagnostic chatbot that combines generation accuracy with retrieved and generative explainability for interactive medical consultations. The system engages users (patients or clinicians) in a dialog, asking relevant follow-up questions, and ultimately providing a diagnostic suggestion with reasoning. MediLink's design prioritizes both technical depth in reasoning and a clear user experience: it can operate in a formal clinical mode for healthcare professionals and a casual chat mode for laypersons, adapting its language and detail accordingly. The system employs a hybrid of methods, including Retrieval-Augmented Generation (RAG) for up-to-date medical knowledge based on a knowledge graph database, LLM-based reasoning for differential diagnosis, and a symptom co-occurrence matrix to design better follow-up questions.

In summary, MediLink offers the following key features and contributions:

- **Multi-Agent Architecture**: MediLink uses a network of specialized modular agents (or modules) that work together, a disease prediction engine with a generative dialogue engine for interactive diagnosis.
- **Standardized Medical Knowledge**: The system maps user-described symptoms and suspected diseases to standard identifiers (ICD-10 codes for diagnoses and UMLS concepts for symptoms) to ensure consistency and enable integration with medical ontologies and terminologies.
- **Dynamic Dialogue with Dual Modes**: It supports both a clinical mode (prioritizing medical terminology and concise communication) and a casual mode (providing patient-friendly explanations and empathetic tone), improving adaptability to different end-users.
- **Retrieval-Augmented Reasoning**: MediLink incorporates external medical literature by querying PubMed via NCBI Entrez APIs. The relevant articles recovered are used to support the final diagnostic reasoning, ground the chatbot's conclusions in up-to-date evidence, and mitigate hallucinations related to LLM [11].
- **High Accuracy and Interpretability**: A fine-tuned diagnostic engine (MedAlpaca-7B [1]) achieves approximately 95.4% accuracy on internal validation cases, and the system's reasoning process is interpretable. It produces a confidence-ranked list of possible conditions and justifies its final recommendation with traceable logic and external references.

The following sections detail the design of MediLink and situate it in the context of related work. We describe the system architecture and methodology, present initial evaluation results, and discuss conclusions and future directions.

Related Work

Early computer-aided diagnosis systems, such as MYCIN and later symptom-checker applications, relied on rule-based inference or Bayesian networks to suggest diagnoses given patient inputs. These systems required manual knowledge engineering and often lacked conversational interfaces. More recent symptom checker platforms employ statistical learning on symptom-disease databases, but they typically follow a fixed questioning script and do not leverage the rich language understanding of modern LLMs.

The use of large language models like GPT-3/GPT-4 has increased interest in AI that can perform differential diagnosis through dialogue (With Conversational Capabilities). For example, researchers have demonstrated that GPT-4 can generate plausible diagnostic lists from patient descriptions, but concerns remain about accuracy, transparency, and the potential for incorrect but confident answers (i.e., hallucinations). To address these issues, one emerging trend is *retrieval-augmented generation (RAG)*: incorporating external knowledge sources or databases into the LLM's reasoning process. Recent studies emphasize augmenting LLMs with medical knowledge graphs or clinical databases to improve diagnostic accuracy and trustworthiness of the system. This has led to systems that can retrieve relevant clinical guidelines or research articles associated with the model's predictions, thus providing evidence for the AI's conclusions.

Another line of research focuses on making the diagnostic reasoning of LLMs more interpretable and stepwise. Chen *et al.* introduced the **Chain-of-Diagnosis (CoD)** approach [10], which transforms the diagnostic process into a transparent sequence of reasoning steps, analogous to a physician's thought process. In their CoD framework, the LLM explicitly outputs a chain of reasoning with identified symptoms, intermediate conclusions, and a confidence distribution over possible diseases. This method helps explain why certain diagnoses are considered or ruled out and identifies which additional symptom inquiries could most reduce uncertainty (using entropy reduction of the confidence distribution). The CoD-based system (DiagnosisGPT) is reported to cover 9,604 diseases and showed superior performance on diagnostic benchmarks [1], highlighting the power of tailored reasoning strategies in medical AI.

MediLink shares the goal of interactive and interpretable diagnosis with these systems, but takes a distinct multi-agent approach. Unlike CoD's single-LLM chain-of-thought method, MediLink delegates tasks to specialized components: one component focuses on symptom/disease identification and confidence estimation, while another handles dialogue generation and inquiry. This design allows MediLink to integrate the retrieval of external information more directly and to tailor its interaction style to the user. Our use of standard medical codes (ICD-10 [8], UMLS [7]) also connects to efforts in medical NLP to ground concepts in controlled vocabularies for consistency and integration with electronic health records. In contrast to end-to-end LLM solutions, MediLink's modular architecture provides a balance between the structured reliability of a knowledge-driven system and the flexibility of a generative model. To our knowledge, few existing diagnostic chatbots combine classification, generative dialogue, and retrieval as explicitly as MediLink does. In the next section, we detail this architecture and how it operates, and how it can respond with more transparency.

Dataset

The dataset utilized for developing and evaluating MediLink comprises associated symptoms and diseases, employed from Huggingface [5]. Initially sourced as raw symptom-disease pairs, the dataset underwent rigorous preprocessing and enrichment to ensure clinical accuracy and interoperability:

Data Preprocessing

The initial dataset was extensively cleaned by removing conversational artifacts, overly lengthy or ambiguous entries, and irrelevant data. Entries were restricted to clearly defined medical conditions, each characterized by concise disease names (a maximum of three words) to ensure clinical clarity and ease of reference.

ICD-10 Code Integration

Accurate disease categorization and clinical interoperability were achieved by integrating ICD-10 codes using the World Health Organization's (WHO) ICD API. An OAuth2-based authentication mechanism was implemented to securely interact with the WHO ICD-10 API, systematically mapping each disease entry to its corresponding ICD-10 code. This mapping process included error handling and retry mechanisms to ensure comprehensive coverage and all the diseases are mapped correctly.

SNOMED-CT Symptom Mapping

Symptoms extracted were standardized by mapping them to the Systematized Nomenclature of Medicine Clinical Terms (SNOMED-CT). A medical Named Entity Recognition (NER) model (en_ner_bc5cdr_md) from SciSpacy was employed to accurately identify and link each symptom to standardized SNOMED-CT terms to handle ambiguity. The final dataset structure differentiates raw symptom descriptions from their standardized SNOMED-CT representations, ensuring consistency and clinical utility.

UMLS Metadata Enrichment

To further enhance clinical relevance and facilitate deeper semantic understanding, additional metadata from the Unified Medical Language System (UMLS) was integrated into the dataset. It was not used in any training, but to have rich metadata for the dataset, we enhanced the dataset from normal SNOMED-CT names to rich metadata. This enrichment involved:

- Fetching concept identifiers (CUIs) for each symptom from the UMLS API, implementing rate limiting and error handling to manage API interactions.
- Extracting detailed semantic information, including semantic types, preferred terms, synonyms (atoms), severity, temporality, and anatomical context.
- Employing advanced natural language processing techniques, including severity and temporality detection, and anatomical context extraction using specialized SciSpacy models.

The resulting enriched dataset encapsulates comprehensive medical information structured to directly support diagnostic modeling, symptom co-occurrence analysis, and clinical reasoning tasks within MediLink. This robust preprocessing and enrichment approach not only enhances the diagnostic accuracy of the system but also lays a solid foundation for potential future integrations with electronic health records and other multimodal clinical datasets.

RAG Model Dataset

A supplementary dataset was employed from huggingface [6] specifically for Retrieval-Augmented Generation (RAG) purposes. This dataset, sourced from the FreedomIntelligence Disease Database, contains detailed entries of diseases, common symptoms, and treatments. The preprocessing steps for this dataset included:

- Extraction and cleaning of symptoms from raw descriptions to ensure clarity and standardization.
- Mapping diseases to corresponding ICD-10 codes using the WHO ICD API, similarly implementing secure OAuth2 authentication and robust error handling.
- Standardizing symptoms using the UMLS API to acquire detailed metadata, including CUIs, semantic types, synonyms, and standardized terminology.
- Further enhancing data interoperability and retrieval effectiveness by generating embeddings using SentenceTransformer models and uploading these embeddings to Pinecone for efficient semantic retrieval.

The integration of this secondary dataset facilitates accurate semantic search capabilities, enriching the MediLink diagnostic process through ensemble scoring methods that improve confidence and diagnostic accuracy.

Ensemble Confidence Scoring

To improve diagnostic robustness, MediLink incorporates ensemble confidence scoring by combining predictions from generation and retrieval-based models. Top-1 and Top-5 predictions from generation models are aggregated with retrieved cosine similarity scores from the RAG module, which are used to validate and rerank final scores. This multi-pronged approach ensures better reliability, particularly for rare diseases and ambiguous inputs.

Table 1: Comparison of Datasets Used

Aspect	Training Dataset	RAG Dataset
Source	Custom-labeled symptom-disease dataset	FreedomIntelligence Disease DB
Symptom Mapping	SNOMED-CT + UMLS Standardization	UMLS Metadata, SNOMED-CT
Disease Codes	ICD-10 via WHO API	ICD-10 via WHO API
Purpose	Prediction	Retrieval
Vector Store	Not applicable	Pinecone Semantic Index
Format	[Symptoms] → [Disease Label]	[Symptoms, Disease, Treatment] Entries

Training and Evaluation

The MediLink system leverages only generation-based architectures, but the task is classification-based; we tested with both approaches to build a robust disease prediction engine. Fine-tuning was carried out using both causal language modeling (CAUSAL_LM) and sequence classification objectives, allowing comparative analysis across multiple model families and configurations.

Model Architectures and Training Modes

We employed MedAlpaca-7B [1] and LLaMA-3.2 [2] series (1B and 3B) models using LoRA-based [3] parameter-efficient fine-tuning. Training was performed using DeepSpeed (Stage 2 ZeRO optimization) on multiple GPUs with bfloat16 precision. Two modes of training were implemented:

- **Generation (Causal LM):** Prompts were formatted as “`### Symptoms: <symptom list> \n\n### Diagnosis:`” with the target disease name used as completion. This allowed open-ended disease generation and ranking via similarity and confidence scores.

- **Classification:** Input prompts were tokenized as in generation, but the models were trained to predict class labels from a fixed disease vocabulary. Class imbalance was handled by deduplication and minimal sample augmentation (min 6 examples per class).

Training Configuration

Most hyperparameters remained consistent across experiments to ensure fair comparison. Key parameters are detailed below:

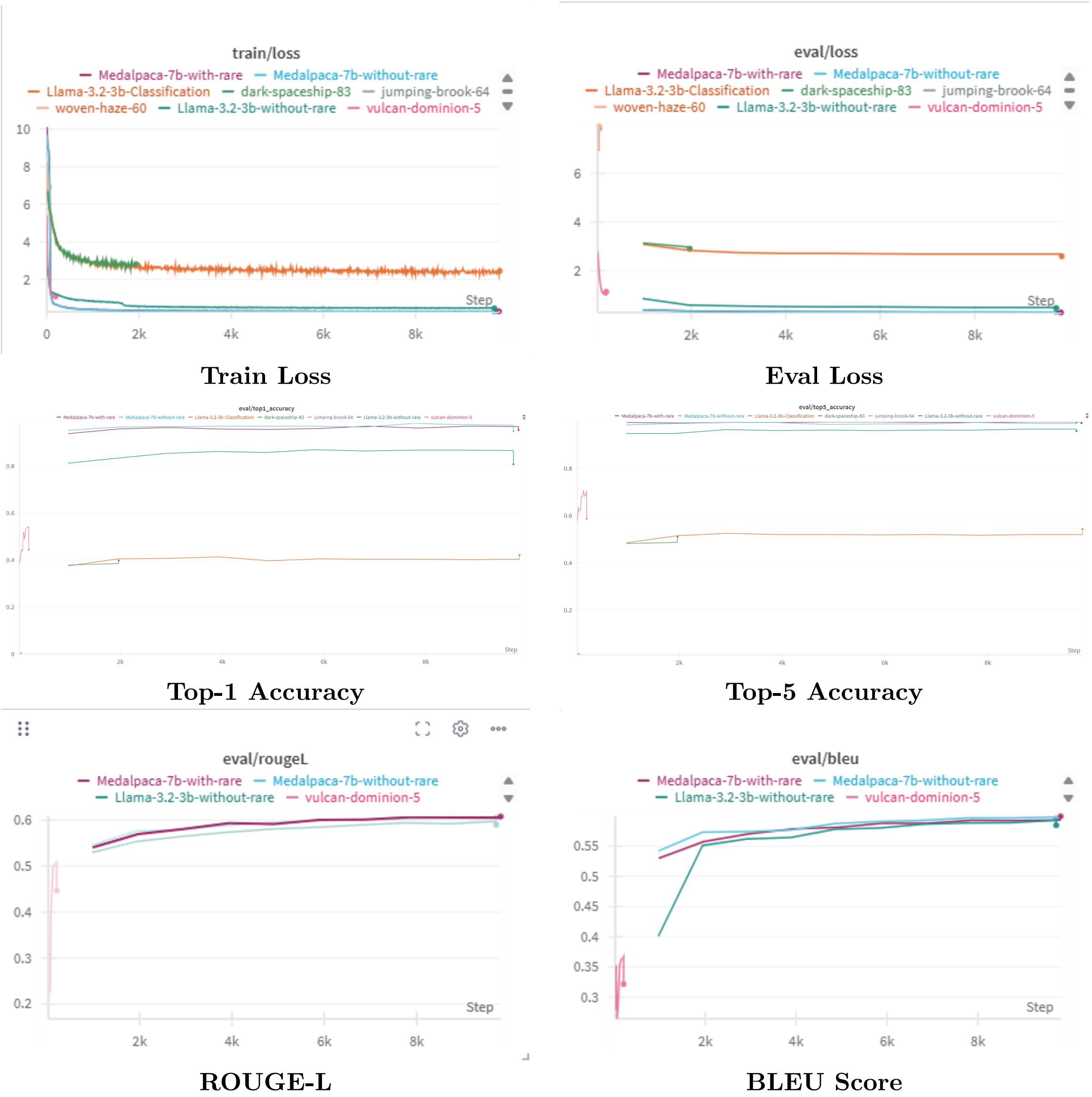
Table 2: Training Hyperparameters

Parameter	Causal LM (Generation)	Sequence Classification
Model	MedAlpaca-7B, LLaMA-3.2 (1B/3B)	LLaMA-3.2-3B
LoRA Rank (\$r\$)	16	16
LoRA \$α\$	32	32
Per-GPU Batch Size	8	8
Learning Rate	\$2 ×10^{-5}\$	\$2 ×10^{-5}\$
Epochs	10	10
Deepspeed ZeRO Stage	2	2
Evaluation Strategy	epoch	epoch

Performance Metrics

Models were evaluated using Top-1 and Top-5 Accuracy, BLEU, ROUGE-L, and generation/classification loss on training and test subsets.

Results are aggregated below:



Train Loss

Eval Loss

Top-1 Accuracy

Top-5 Accuracy

ROUGE-L

BLEU Score

Figure 1: Training and evaluation metrics across model variants

Table 3: Final Performance Comparison

Model Variant	Top-1 Acc.	Top-5 Acc.	BLEU	ROUGE-L	Eval Loss
MedAlpaca-7B (w/ rare)	**0.954**	**0.993**	0.599	0.608	0.28
MedAlpaca-7B (no rare)	0.948	0.992	0.595	0.607	0.30
LLaMA-3.2-3B (Gen)	0.808	0.958	0.584	0.590	0.46
LLaMA-3.2-3B (Cls)	0.421	0.543	-	-	2.59

Observations

- **MedAlpaca-7B with rare disease augmentation outperformed all other variants** across metrics, validating the importance of rare disease representation.
- **Classification-only models (LLaMA-3.2-3B Cls)** showed significantly lower performance due to fixed vocabulary constraints and lack of generation flexibility.
- **We selected MedAlpaca-7B as the final model** based on its superior Top-1 and Top-5 accuracy on the test set, achieving over 95% accuracy even with rare diseases included.
- BLEU and ROUGE-L scores correlated strongly with Top-1 accuracy in generation tasks, validating their utility as surrogate measures for correctness.

Training loss, evaluation loss, and accuracy curves are visualized in the figures above to highlight convergence dynamics and performance plateaus.

Methodology (System Architecture)

MediLink uses a sophisticated, multi-layered architecture designed for detailed processing of user inputs, systematic diagnostic hypothesis generation and refinement, and comprehensive evidence-based medical reasoning. Figure 2 clearly illustrates the interactions and data flow between the four primary layers: User Interaction, Core Diagnostic Loop, Knowledge and Evidence Retrieval, and Reasoning.

User Interaction Layer

The user interaction is managed via an intuitive conversational chatbot interface, implemented using the Gradio interface. This interface accepts both text and speech inputs from a user, enhancing usability across diverse user groups, Currently, it handles English text and speech. Upon initial engagement, the user inputs their symptoms freely in raw text, after which the system checks and collects necessary demographic details if missing, such as age, sex, weight, and height—through structured prompts managed by a dedicated Session Orchestrator. The orchestrator maintains conversational state, ensuring seamless transitions between diagnosis states, demographic data collection, and conversational interactions.

Core Diagnostic Loop

The central diagnostic loop encompasses multiple critical steps:

Symptom Extraction and Standardization Symptoms described by the user in natural language are accurately extracted using OpenAI's GPT-3.5 Turbo API [13]. Extracted symptoms are standardized to structured terms referencing the Unified Medical Language System (UMLS) [7] to maintain consistency and facilitate downstream processes.

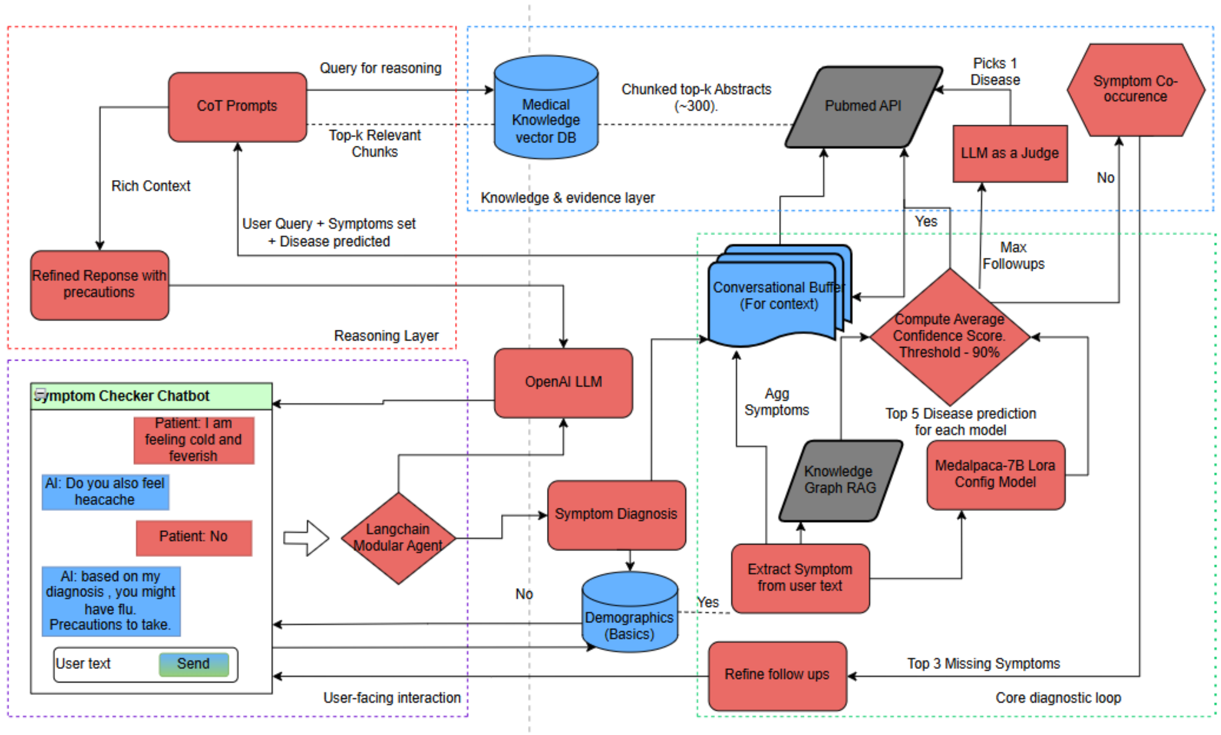

Query for reasoning
CoT Prompts
Top-k Relevant Chunks
Medical Knowledge vector DB
Chunked top-k Abstracts (~300).
Pubmed API
Picks 1 Disease
LLM as a Judge
Symptom Co-occurence
No
Rich Context
Knowledge & evidence layer
User Query + Symptoms set + Disease predicted
Yes
Max Followups
Refined Reponse with precautions
Conversational Buffer (For context)
Compute Average Confidence Score. Threshold - 90%
Reasoning Layer
OpenAI LLM
Symptom Checker Chatbot
Patient: I am feeling cold and feverish
AI: Do you also feel heacache
Patient: No
AI: based on my diagnosis , you might have flu. Precautions to take.
User text
Send
Agg Symptoms
Top 5 Disease prediction for each model
Knowledge Graph RAG
Medalpaca-7B Lora Config Model
Langchain Modular Agent
Symptom Diagnosis
No
Demographics (Basics)
Yes
Extract Symptom from user text
Refine follow ups
Top 3 Missing Symptoms
User-facing interaction
Core diagnostic loop

Figure 2: MedLink System Architecture

Parallel Disease Prediction MediLink employs a dual-engine diagnostic prediction system:

- **Retrieval-Augmented Generation (RAG)**: Uses embeddings created from user-reported symptoms to retrieve related diseases from a Pinecone [15] vector database. This database has been populated previously with high-quality symptom-disease mappings.
- **Generative Diagnostic Model**: Implements a fine-tuned MedAlpaca-7B model via Low-Rank Adaptation (LoRA). This model directly predicts a ranked list of diseases based on provided symptoms, and leverages beam search over disease prediction. The reason for choosing generation over classification is provided in the training section.

Confidence Evaluation and Ensemble Prediction The outputs from both the RAG and generative models are assessed through dynamic weighting based on prediction confidence levels. The system dynamically adjusts the weights (For e.g., 70% RAG and 30% LLM if high confidence in LLM predictions, otherwise a balanced 50-50 split) to produce a combined disease ranking and decision-making.

Symptom Co-occurrence and Follow-Up Question Generation When the confidence score is below the defined threshold (80%), MediLink employs a symptom co-occurrence analysis module implemented using disease-symptom frequency and co-occurrence counters. This module identifies missing critical symptoms based on retrieved and predicted top-5 diseases from RAG and LLM, each of which, if clarified, could significantly reduce diagnostic ambiguity and enhance the system's confidence score. Follow-up questions targeting these symptoms are dynamically generated via OpenAI's GPT-4o API, tailored according to current confidence levels—fewer and more specific questions at high confidence, broader inquiries at lower confidence levels.

Iterative Refinement Loop MediLink iteratively repeats the follow-up questioning and symptom clarification process until the confidence score exceeds 0.80 or until the maximum allowable follow-ups (in our case, we limited it to three) are reached.

Knowledge and Evidence Retrieval Layer

Once a top candidate disease is selected from the diagnostic loop, MediLink queries external medical knowledge sources using the PubMed API (via Entrez) based on the patient's demographics, user interactions to understand context, the set of symptoms extracted in a diagnostic loop, and the final disease prediction. Relevant abstracts fetched from PubMed are segmented into smaller chunks (approximately 300 words each) and stored as embeddings in a medical knowledge vector database (ChromaDB). This database facilitates fast retrieval of contextually relevant medical evidence.

Reasoning Layer

This final layer synthesizes aggregated data from the user and evidence into a coherent medical explanation using a structured Chain-of-Thought (CoT) reasoning process:

Context Aggregation The system combines comprehensive contextual information, user demographics, complete conversational history, extracted symptoms, and diagnostic predictions, to inform the reasoning process.

Evidence-Based Chain-of-Thought Reasoning MediLink retrieves the most relevant medical knowledge chunks from ChromaDB based on the aggregated context. These evidence chunks are then systematically processed through OpenAI's GPT-4o, producing detailed reasoning that includes:

- **Pathophysiological Explanation**: Biological and medical rationale behind the suspected diagnosis.
- **Diagnostic Criteria**: Alignment with established diagnostic guidelines.
- **Symptom Match Assessment**: Evaluation of symptom relevance and specificity.
- **Differential Diagnosis**: Consideration and justification of alternative diagnoses.
- **Evidence Evaluation**: Critical analysis of supporting medical literature.
- **Final Confidence Assessment**: Explicit declaration of confidence in the final diagnostic conclusion.

Final Response Generation The output reasoning is condensed into an accessible summary, clearly outlining the diagnosis, recommended treatments (Fetched from RAG Database), and essential precautions to take. Each response explicitly includes a disclaimer emphasizing the need for professional medical consultation, as predicted disease is within the limited scope of the database, and in some cases, it might miss the critical information.

Supporting Modules and Utilities

Several specialized modules enhance MediLink's diagnostic and conversational capabilities:

- **ICD-10 Mapper**: Utilizes WHO's ICD API for accurate mapping of disease predictions to standardized ICD-10 codes, ensuring clinical interoperability.
- **LangChain Modular Agent**: An intent classifier built using LangChain's ZeroShotAgent, categorizing user inputs into symptom diagnosis requests, patient history inquiries, or general conversational interactions.
- **Vector Memory Stores**: Dedicated ChromaDB instances manage conversational memory (prior interactions), structured metadata, and medical knowledge separately, thus optimizing the system's retrieval efficiency and contextual accuracy.

MediLink's robust, modular architecture is specifically designed for interpretability, extensibility, transparency, reliability, and maintainability. This comprehensive design facilitates independent improvements, potential integration with electronic health records, multimodal data sources, and advanced reasoning methodologies in future iterations.

Conclusion

We presented MediLink, a multi-agent medical diagnostic chatbot that integrates the strengths of generative algorithms and retrieval augmented generation. By structuring the diagnostic process into modular steps (symptom parsing, hypothesis generation, iterative inquiry, and evidence-backed explanation), MediLink provides both accuracy and interpretability. The system leverages standard medical ontologies (ICD-10 codes and UMLS concepts) to maintain a consistent understanding of medical terms throughout the interaction, which is crucial for integrating with electronic health records or decision support tools. Through a dynamic dialogue, it intuitively engages users, asking relevant follow-up questions much like a human physician would.

A major feature of MediLink is its ability to adapt responses based on the audience: healthcare professionals receive a concise, terminology-rich explanation with references, whereas patients receive a compassionate and detailed explanation. This dual-mode communication aims to maximize the usefulness of the system in both clinical settings (as a decision support or training tool) and direct-to-consumer health advice settings (as a preliminary triage or informational service). The inclusion of Retrieval-Augmented Generation ensures that MediLink's knowledge remains current and its explanations can be audited against reputable sources. This addresses one of the significant limitations of generative models, the tendency to produce outdated or unsupported statements, by anchoring the final output in published medical literature.

In conclusion, MediLink demonstrates that a carefully designed combination of AI techniques can yield a robust interactive diagnostic assistant. It shows that LLMs, when constrained and guided by structured medical knowledge and supplemented with retrieval, can achieve impressive diagnostic reasoning performance (comparable to specialized models [10]) while also delivering user-centric communication. We believe such systems can serve as valuable adjuncts in healthcare: assisting clinicians by double-checking diagnoses and providing evidence, and helping patients by interpreting their symptoms and encouraging appropriate follow-up. The modular architecture of MediLink will also facilitate future enhancements as the field of medical AI evolves.

Future Work

While the current MediLink prototype is effective, there are several areas for future improvement and expansion. First, we plan to integrate **temporal reasoning and symptom severity** into the diagnostic process. Real clinical scenarios often involve symptoms changing over time (e.g., a fever that peaked and then broke, or pain that is gradually worsening). We aim to allow users to describe symptom timelines and severity levels (mild, moderate, severe), and adjust the diagnostic reasoning accordingly. Incorporating temporal patterns could help differentiate diseases (for instance, intermittent fevers vs. continuous fevers can suggest different etiologies).

Second, we will incorporate **patient medical history and risk factors** into MediLink's reasoning. This involves extending the input beyond just current symptoms: the system should consider chronic conditions (like diabetes or hypertension), past surgeries, medications, allergies, and family history. Such factors heavily influence diagnostic probabilities in medicine. We may integrate with standards like FHIR for structured health records to pull in a user's history (with permission), or allow the user to input key history elements conversationally. The diagnostic engine and co-occurrence models would then need to condition on this information (for example, chest pain in a patient with a history of coronary artery disease warrants different suspicion than in a young healthy patient).

Third, we plan to leverage **deeper UMLS metadata and knowledge graph relationships**. Currently, we use UMLS primarily for identifying concept IDs of symptoms and diseases. In the future, we can utilize the rich semantic types and relationships in UMLS (and other knowledge sources like SNOMED CT) to enhance reasoning. For example, knowing the anatomical location of a symptom (UMLS semantic type for body location) could help MediLink ask more anatomically focused questions or rule out diseases that don't match the location. Semantic relations (such as ”finding site”, ”associated with”) could allow the system to navigate a knowledge graph of disease-symptom-test relationships. This could enable suggestions like recommending certain diagnostic tests or considering alternative diagnoses that are related via shared findings. We also intend to use UMLS to better handle synonymy and variant phrasing beyond what the current NER covers, so that the system is more robust to different ways patients might describe the same concept.

In addition to these specific enhancements, ongoing future work will include:

- **Continuous Learning**: Enabling MediLink to learn from new interactions (with appropriate oversight), gradually expanding its knowledge of rare conditions or atypical presentations by incorporating new case data, but limiting these features to only medical professionals to make sure unnecessary knowledge is not added.
- **Evaluation and Safety**: Conducting more extensive evaluations with clinicians and patients, and assessing the safety of the advice given. We plan to implement a feedback loop where medical experts can review and correct MediLink's outputs, helping to refine the system. Furthermore, we will incorporate safety checks to recognize situations where the system should advise urgent medical attention or defer to human professionals.
- **User Interface and Deployment**: Improving the front-end interface for MediLink, such as a smartphone app or web portal, with features like voice input/output for accessibility. We also plan integration with electronic health record systems in clinical mode, so that a clinician can seamlessly use MediLink during patient visits and log the AI's suggestions and literature citations for reference.
- **Multimodal Integration**: Exploring adding capabilities for MedLink to handle other data types, such as basic lab results or images. For instance, linking with a skin lesion image classifier agent or a lab test interpreter could extend the system's diagnostic range (this would effectively add more specialized agents to the architecture).

By addressing these areas, we hope to make MediLink an even more comprehensive and reliable diagnostic assistant. The combination of temporal data handling, patient history integration, and semantic knowledge graph use will move the system closer to how a human doctor thinks—considering the whole patient, not just isolated symptoms. We anticipate that these future improvements will further bridge the gap between AI recommendations and real-world clinical decision making, ultimately contributing to safer and more effective use of AI in healthcare.

References

[1] T. Han et al., "MedAlpaca: An Open-Source Collection of Medical Conversational AI Models and Training Data," *arXiv:2304.08247*, 2023.

[2] H. Touvron et al., "LLaMA: Open and Efficient Foundation Language Models," *arXiv:2302.13971*, 2023.

[3] E. J. Hu et al., "LoRA: Low-Rank Adaptation of Large Language Models," *ICLR*, 2021.

[4] P. Lewis et al., "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," *NeurIPS*, 2020.

[5] Akhilesh , "Symptom Checker", *HuggingFace Datasets*, 2024. [Online]. Available: `https://huggingface.co/datasets/akhileshav8/symptom_checker`

[6] FreedomIntelligence Team, "Disease-Symptom-Treatment Knowledge Graph," *HuggingFace Datasets*, 2022. [Online]. Available: `https://huggingface.co/datasets/FreedomIntelligence/Disease_Database`

[7] O. Bodenreider, "The Unified Medical Language System (UMLS): Integrating Biomedical Terminology," *Nucleic Acids Research*, vol. 32, pp. D267–D270, 2004.

[8] WHO, "International Statistical Classification of Diseases (ICD-10)," 10th Revision, 2019.

[9] SNOMED International, "Systematized Nomenclature of Medicine–Clinical Terms (SNOMED-CT)," Technical Specification, 2023.

[10] J. Chen et al., "CoD: Towards an Interpretable Medical Agent using Chain of Diagnosis," *arXiv:2407.13301*, 2024.

[11] S. Zhou et al., "Large Language Models for Disease Diagnosis: A Scoping Review," *arXiv:2409.00097*, 2024.

[12] K. Singhal et al., "Towards Expert-Level Medical Question Answering with Large Language Models," *arXiv:2305.09617*, 2023.

[13] T. Wolf et al., "Transformers: State-of-the-Art Natural Language Processing," *EMNLP*, 2020.

[14] LangChain Authors, "LangChain Framework," 2023. [Online]. Available: `https://python.langchain.com`

[15] Pinecone Systems, "Vector Database for AI Applications," 2023. [Online]. Available: `https://www.pinecone.io`

[16] Y. Liu et al., "Evaluating the Clinical Utility of AI-Assisted Symptom Checkers," *JAMA Network Open*, vol. 6, no. 5, 2023.