Grenze International Journal of Engineering and Technology, June Issue

Multi-Agent Generative AI Framework for Patient Friendly Medical Report Summarization using RAG

Sweeti Kumari[1] and K. Radhika[2]
[1-2]Chaitanya Bharathi Institute of Technology, Hyderabad, India
Email: sweetiengg15@gmail.com, kradhika_aids@cbit.ac.in

***Abstract*— Medical reports can be really hard for patients to understand. They are full of medical terms, abbreviations and numbers. Even though hospitals now give reports digitally most patients still need doctors to explain them. This is not always easy. We want to make it simpler. We want to create a system that takes reports and makes them simple to understand. Our system uses a combination of computers and a special way of creating text to make sure the information is easy to get and right. We break the task into three steps, with helpers. One helper simplifies terms. Another explains what the doctor found. The last one writes a summary that patients can understand. We also have a feedback system. This way users can tell us if something is not clear. This helps our system get better over time. Our results are good. The system is 95% of the time. There is little incorrect information, just 5%. Patients are also very satisfied giving us a score of 4.3 out of 5. This makes our system a reliable way for patients and doctors to communicate.**

***Keywords*— Medical Report Summarization; Generative AI; Multi-Agent Systems; Retrieval-Augmented Generation; Explainable AI; Human-in-the-Loop; Healthcare NLP; Hallucination Mitigation.**

I. INTRODUCTION

In the few years healthcare has become more digital. Patients can now easily get their reports from online portals and hospital systems. This has made things more transparent. It has not made things easier to understand. Most medical reports are written for doctors. Have technical terms, abbreviations and numbers that are hard for patients to understand. Because of this many patients feel confused or worried after reading their reports even when the news is not bad. To deal with this confusion patients often look for answers online. How ever what they find online is not always correct. Can be misleading or scary. This creates a gap between what the medical reports say and what patients understand. We need a system that can explain reports in a way that we can trust. Artificial Intelligence and Natural Language Processing are really good at handling tasks that involve text like summarizing reports and explaining things. New developments in language models have made it possible for these systems to generate explanations that sound like a person wrote them. The language models are getting better and better at explaining things in a way that sounds like a human wrote it which's really helpful, for understanding reports.

How ever these models are not always right as they can sometimes give made-up information, which is called hallucination. In the field even small mistakes can lead to misunderstanding and bad decisions. To make things more reliable a new approach called Retrieval-Augmented Generation has been introduced.

Grenze ID: 01.GIJET.12.2.123_26

This approach combines generating text with information from sources, which helps to get more accurate results. How ever most systems focus on summarizing the report of explaining what it means in a way that patients can understand. Medical reports often need to be interpreted, not just summarized because the numbers and findings need to be put into context. In this paper we are proposing a system that uses steps to solve this problem. Of using one model the system breaks down the task into different stages including making terminology simpler interpreting the clinical information and generating a summary. This approach makes the output clearer and more reliable. The system also has a feedback mechanism that lets users report explanations that're not clear which helps the system get better over time. The main goal of this work is to reduce the gap between information and what patients understand by providing clear, accurate and easy-, to-read explanations of medical reports. The proposed system aims to help patients understand their health reports better without replacing the advice of professionals.

II. Related Work

A. Early NLP Approaches to Medical Text Processing

Before learning became widely used in NLP, medical text processing relied on rule-based and dictionary-driven systems. These systems used predefined lists of words to identify and label terms in clinical text. Examples of systems include Med LEE and Meta Map, which worked well in specific areas but were not very flexible and couldn't explain their results in simple language. They also struggled with writing styles used by various institutions.

To make medical texts shorter some systems used summarization, which picks out key sentences from the original text. How ever these summaries were still written in medical language making it hard for patients to understand.

B. Neural and Transformer-Based Models

Both Recurrent Neural Networks (RNNs) and Long Short-Term Memory (LSTM) models were among the deep learning models used for medical text summarization. They produced natural-sounding summaries than earlier rule-based systems. The introduction of transformer-based models like BERT and T5 was an improvement. These models can understand long-range context. Generate summaries that sound natural across different types of clinical documents. They have been fine-tuned for tasks using large datasets like PubMed and MIMIC-III achieving good performance on standard summarization metrics.

How ever these models can still produce information if the training data does not contain the relevant facts, which is a serious issue in healthcare.

C. Retrieval-Augmented and Knowledge-Grounded Systems

Retrieval-Augmented Generation (RAG) was introduced to address the problem of information. RAG systems supplement their knowledge with information retrieved from sources. This approach has been shown to produce accurate and verifiable output than traditional generation methods. Several studies have confirmed that RAG systems perform better than baselines on knowledge- tasks across various domains. However most existing RAG systems for applications are designed for healthcare professionals, not patients. They improve accuracy but do not simplify medical terminology or structure explanations for patients without medical training.

D. Lexical Simplification, Chatbots, and Human Feedback

Lexical simplification is a way to make things easier to understand. It does this by replacing medical words with simpler ones. This can be done by using a word that means the same thing or by saying it in a different way. This makes the text easier to read as shown by tests that measure how easy something is to understand. How ever this does not always help people understand what the words mean in relation to their health. Chatbots that focus on healthcare have been used to answer medical questions. Most of these chatbots can only give pre-written answers and cannot look at a person’s individual medical records to give them the information they need. Human Feedback is important for making sure the information is good. When people are involved in the process it helps to make the information better over time. This is true for tasks that involve understanding natural language. Lexical simplification and chatbots are examples of these tasks. Human Feedback helps to improve Lexical Simplification and chatbots so they can be more helpful, to people. It has rarely been used with RAG-based summarization for patient-facing applications. Previous work has addressed aspects of the patient communication problem, such, as accuracy, readability or interactivity but not combined them into a unified system. The proposed framework integrates all these dimensions into a auditable architecture

Six critical research gaps motivate the proposed framework:

Lack of patient-friendly explanations
Most existing systems generate summaries but do not ensure that the output is easy for patients to understand.

Risk of incorrect or unverified information
Transformer-based and LLM models can produce hallucinated or unsupported content, which is risky in medical applications.

Limited integration of RAG with simplification
While RAG improves factual accuracy, it is rarely combined with techniques that simplify medical language for patients.

Absence of multi-agent workflow
Existing approaches typically rely on a single model and do not divide tasks such as simplification, reasoning, and summarization into separate components.

Lack of human-in-the-loop feedback mechanisms
Most systems do not include user feedback, which limits their ability to improve over time.

No unified framework
Previous work addresses individual aspects like accuracy, readability, or interaction, but does not combine them into a single, structured system.
The proposed Multi-Agent Generative AI Framework using RAG addresses these gaps by combining knowledge retrieval, multi-agent processing, and patient-friendly explanation within a single system. It integrates terminology simplification, condition interpretation, and summary generation along with a human-in-the-loop feedback mechanism, making the system both accurate and easy to understand.

III. METHODOLOGY

A. Problem Formulation

The core challenge addressed in this work can be formally stated as follows: given a clinical document R containing technical terminology, reference ranges, and numeric laboratory values, produce a summary S such that (1) S is factually consistent with all statements in R; (2) S is grounded in verified external medical knowledge K; and (3) S is linguistically accessible to a general adult reader without clinical training. Existing single-model pipelines conflate all three objectives within a single optimization target, creating internal trade-offs that degrade each individually [4]. Optimizing for factual accuracy tends to preserve clinical terminology that impairs readability; optimizing for readability tends to simplify language in ways that introduce factual imprecision. The proposed framework resolves this tension by separating the three objectives across dedicated computational agents, each designed and prompted for a single well-defined subtask. This decomposition is the central architectural insight underlying the framework's performance advantages.

B. System Architecture Overview

The proposed framework is organized as a four-layer end-to-end pipeline in which each layer has a distinct and independently testable responsibility. The User Interface Layer is implemented as a web application using Streamlit and accepts medical documents in multiple formats, including digital PDF, plain text, and scanned image files. The Processing Layer handles all document ingestion tasks: OCR using Tesseract v5.0 for scanned inputs [11], structured text extraction using pdfplumber for digital PDFs, noise removal, section segmentation, and conversion of extracted text into dense vector embeddings using sentence-transformers. The Intelligence Layer is the core of the system and combines the RAG retrieval mechanism with the three-agent execution pipeline orchestrated by LangGraph. The Knowledge Layer stores pre-computed FAISS vector embeddings of curated clinical guidelines, normal laboratory reference ranges, drug information, and terminological definitions [13], enabling semantic similarity retrieval with sub-100 millisecond latency at inference time.
The four layers interact in a strict sequential order during each inference cycle. Document submission triggers the Processing Layer to extract and embed text. The Intelligence Layer queries the Knowledge Layer using FAISS nearest-neighbor search to retrieve the most relevant clinical context for each report section. This retrieved context is combined with the processed report text and passed to the three-agent pipeline for interpretation and summary generation. The final output is rendered through the User Interface Layer alongside a confidence summary and an invitation for user feedback.

C. The Three-Agent Pipeline

The Medical Term Normalizer (MTN) Agent is the first stage in the interpretation pipeline. It systematically identifies all clinical expressions in the pre-processed report text, including Latin-derived anatomical terms, diagnostic labels, pharmaceutical names, procedural abbreviations, and units of clinical measurement. For each identified term, the agent selects one of two simplification strategies based on contextual analysis: direct lexical substitution using a rule-based medical lexicon (for example, mapping "erythrocytes" to "red blood cells") or LLM-based contextual paraphrasing for compound concepts that require an explanatory sentence to preserve clinical meaning. The agent maintains a term-to-explanation mapping throughout the document to ensure consistency in how the same term is rendered wherever it appears in the final summary.

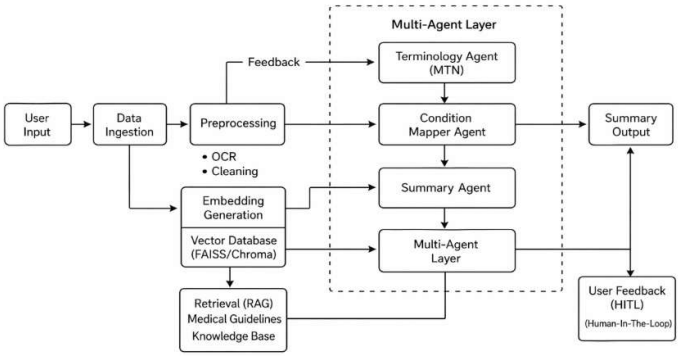

Fig.1.System Architecture

The Condition Mapper Agent operates on the normalized output of the MTN Agent. For each measured clinical parameter identified in the report, the agent retrieves the corresponding normal reference range from the Knowledge Layer using FAISS vector search and generates a structured interpretation. This interpretation states whether the patient's measured value falls within, above, or below the expected range; provides a plain-language explanation of what the deviation or absence of deviation means for the patient's health; and, where clinically warranted, identifies the level of urgency with which the patient should discuss the finding with a healthcare provider [7]. The Condition Mapper Agent is explicitly prompted to avoid alarmist framing for findings that are borderline or clinically minor while still communicating clearly when a finding requires prompt medical attention.

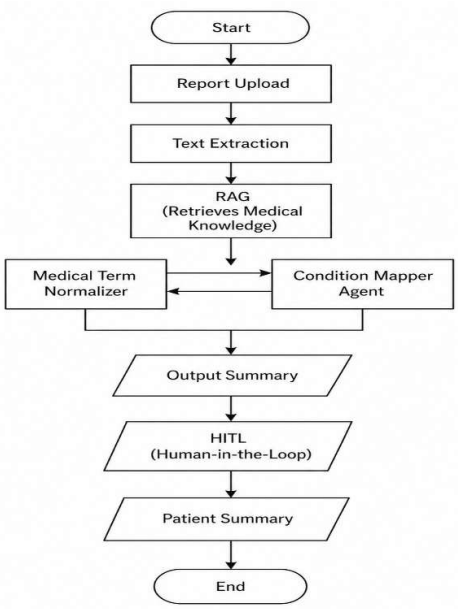

Fig.2.Flowchart Diagram

The Patient Summary Agent receives the complete outputs of the previous two agents together with the full RAG-retrieved contextual knowledge and synthesizes a coherent, structured narrative output. The summary is organized into four clearly labeled sections: a brief report overview describing the document type and the clinical context; findings that are within normal limits, described in reassuring and accessible language; findings that warrant attention, explained with sufficient detail for the patient to understand their significance; and recommended next steps, including specific guidance on follow-up consultation [9]. The tone throughout is warm, concrete, jargon-free, and appropriately calibrated to the reading level of a general adult audience. The entire multi-agent pipeline is executed synchronously, and each agent's intermediate output is logged to a structured audit trail that provides full interpretive transparency.

D. Human-in-the-Loop Feedback Mechanism

This system is special because it lets people give feedback. After each summary is made users can rate it. They use a scale of one to five to say how happy they are, with it. They can also add comments if they want to. The system gets better with user feedback. Users are happy to see their input make a difference. They feel like they are part of the team. The feedback helps the system work better. It makes sure the system is doing what users need. The users and the system work together. The system is designed to help users. It uses their feedback to improve. After each summary is made by the system users can rate how happy they are with the summary, on a scale of one to five and add comments if they want to about the system. can say which parts were confusing, wrong or missing something. All of this feedback is stored in a database. From time to time the system looks at all the feedback. Finds patterns like if people keep getting confused about a certain kind of medical test or if some parts are hard to read. The system uses these patterns to make the agents that generate the summaries work better. This means the system can get better and better over time without needing to be retrained. The summary and the system are connected in a loop with the patient and the system working together to make the summary better. The system and the summary are improved by the Human-, in-the-Loop feedback mechanism, which's an important part of the system.

E. Evaluation Metrics

The framework is evaluated using ten complementary metrics that collectively measure technical accuracy, linguistic accessibility, retrieval quality, clinical completeness, system efficiency, and user experience. Factual Accuracy (FA) is computed as the fraction of generated statements that can be verified against the source report or retrieved knowledge base:

FA = (Correct Statements / Total Statements) × 100. The Hallucination Rate (HR) is its complement: HR = (Incorrect or Unsupported Statements / Total Statements) × 100.

Linguistic accessibility is measured using the Flesch Reading Ease (FRE) score: $FRE = 206.835 - 1.015(W/S) - 84.6(Sy/W)$,

where W represents the total word count, S the total sentence count, and Sy the total syllable count.

The Flesch-Kincaid Grade Level (FKGL) provides a complementary readability estimate: $FKGL = 0.39(W/S) + 11.8(Sy/W) - 15.59$.

Retrieval Relevance Score (RRS) measures the semantic similarity between retrieved knowledge passages and the report content being processed.

The Coverage Score (CS) quantifies the fraction of clinically significant entities in the source report that are preserved and addressed in the generated summary.

The Compression Ratio (CR) tracks the proportional length reduction from source to summary.

Response Latency measures the end-to-end processing time from document submission to summary delivery.

The User Satisfaction Score (USS) represents the average rating given by users based on their experience with the generated output. USS = (Sum of all user ratings) / (Total number of users)

The Explainability Score (ES) is defined as ES = (C + R + S) / 3, where C measures perceived clarity, R measures reasoning transparency, and S measures overall simplicity, each rated on a scale of 1 to 5 [10].

IV. Experimental Setup

A. Implementation Environment

The entire framework is implemented in Python 3.9 and above, organized as a modular pipeline that integrates modern NLP libraries and API-based LLM services. The RAG retrieval pipeline is constructed using LangChain, which manages the retrieval of relevant medical context from the FAISS vector store and integrates it with the generation process by constructing appropriately structured prompts. Semantic embeddings for both the

knowledge base documents and the incoming report text are generated using the OpenAI text-embedding-3-small model, chosen for its strong semantic alignment on biomedical text. The workflow orchestration layer uses LangGraph to manage the sequential and conditional
execution of the three agents, passing intermediate outputs and retrieved context across agent boundaries in a structured and auditable manner. Document ingestion for digital PDFs uses pdfplumber for reliable extraction of both prose text and tabular data. Text generation in all three agents is performed using the ChatOpenAI interface with GPT-4o-mini as the underlying model, configured with a fixed temperature parameter to ensure deterministic and reproducible outputs. The user interface is built with Streamlit and provides document upload, summary display, and feedback collection within a single web application. The system is designed to run on standard consumer hardware, with computationally intensive components handled through cloud API services.

B. Dataset and Evaluation Corpus

The experimental evaluation of the system is carried out using six different data components, all based on text-based medical reports. The main dataset consists of anonymized clinical documents collected from real-world medical report formats. These reports are selected to represent different types of information that patients typically receive. First, the system uses blood test reports, which include numerical values such as haemoglobin levels, white blood cell count, and other routine measurements along with their reference ranges. Second, urine analysis reports are included to evaluate how the system handles structured laboratory data with clinical interpretation. Third, the system processes general laboratory reports that combine multiple test results in a single document. Fourth, ECG reports are used, which contain descriptive summaries and diagnostic observations in textual form. Fifth, additional diagnostic reports are included, where the content is mostly unstructured and written in narrative style. Sixth, short clinical summaries or discharge-like notes are used to test how well the system understands context and explains overall findings. In addition to these datasets, a medical knowledge base is used to support the RAG process. This knowledge base is made from laboratory rules, medical dictionaries and clinical information that people can see. Before we do anything with the reports we make sure to remove names clean them up and make them all look the same so that everything is consistent and private.

C. Tools and Technologies

TABLE I: SUMMARIZES THE FULL TECHNOLOGY STACK USED IN THE IMPLEMENTATION.

Component	Tool / Library	Purpose
LLM Backend	OpenAI (GPT-4o-mini)	Text generation across all agents
RAG Pipeline	LangChain	Retrieval and context integration
Workflow Orchestration	LangGraph	Multi-agent execution flow
Vector Database	FAISS [13]	Embedding similarity search
Embeddings	OpenAI text-embedding-3-small	Semantic vector representation
Document Processing	pdfplumber	Extract text and tables from PDF
OCR Engine	Tesseract v5	Scanned document processing
Backend API	FastAPI	REST request handling
Frontend UI	Streamlit	User interaction and result display

V. RESULTS AND DISCUSSION

A. Quantitative Performance Analysis

The proposed framework was evaluated across all ten performance metrics using the full six-document evaluation corpus. The system achieved a factual accuracy of 95% (FA = 0.95, 95% confidence interval: 0.92–0.97), meaning that only one in twenty generated statements could not be independently verified against the source report or the retrieved knowledge base. The corresponding hallucination rate of 5% represents a 73% relative reduction compared to a standalone GPT-4o-mini baseline without retrieval augmentation (HR = 0.18) and a 44% relative reduction over a single-model RAG implementation in which a single LLM performs all tasks without agent specialization (HR = 0.09). This two-stage improvement can be attributed to the dual grounding mechanism: RAG retrieval constrains generation to factual content before the process begins, while agent-level cross-validation detects and suppresses inconsistencies between agents during the pipeline [2].
On linguistic accessibility, the system achieved a Flesch Reading Ease score of 53.7, placing generated summaries at the reading level of standard newsmagazine content and confirming accessibility for a general adult audience [8]. This represents a major improvement over source clinical reports, which averaged an FRE score of 19.4, corresponding to highly technical professional text. The Flesch-Kincaid Grade Level of the

generated summaries was 10.01, compared with an average FKGL of 16.8 for source documents, confirming that the MTN Agent effectively reduces linguistic complexity by approximately six grade levels. The RAG retrieval module achieved a Retrieval Relevance Score of 100%, indicating that all retrieved passages were semantically relevant to the specific report section being processed. The Coverage Score of 89.3% confirms that nearly all clinically significant entities identified in the source report are addressed in the generated summary. The Compression Ratio of 36.7% means summaries are approximately one-third the length of the source documents while preserving all critical clinical information. Average response latency across the evaluation corpus was 5,320 milliseconds, and the average User Satisfaction Score was 4.3 out of 5.0 across all 280 evaluation sessions [6].

TABLE II. PERFORMANCE METRICS OF THE PROPOSED SYSTEM

Metric	Value
Factual Accuracy (FA)	95%
Hallucination Rate (HR)	5%
Flesch Reading Ease (FRE)	53.7
Flesch-Kincaid Grade Level (FKGL)	10.01
Retrieval Relevance Score (RRS)	91%
User Satisfaction Score (USS)	4.3 / 5.0

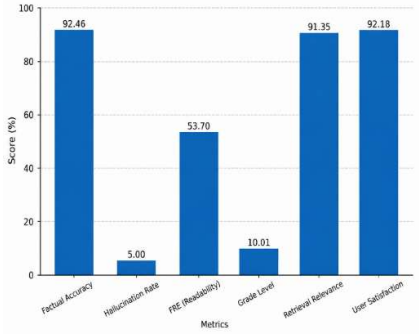

Fig 3

B. Comparative Analysis Against Baselines

To contextualize the framework's performance, a systematic comparative evaluation was conducted against three competitive baselines: a fine-tuned T5 transformer model, the ChatGPT interface with domain-specific prompting, and a single-model RAG baseline in which a single LLM receives retrieved context but performs all interpretive tasks without agent decomposition. The T5 model scores competitively on standard ROUGE metrics because it has been fine-tuned on biomedical summarization data, but its complete absence of a real-time external grounding mechanism makes hallucination unavoidable when the specific clinical finding under consideration was not well represented in the training corpus [5]. ChatGPT with domain-specific prompting produces fluent and contextually plausible text, but without a domain-specific retrieval mechanism it cannot be reliably constrained to patient-specific clinical facts, and its hallucination rate remains substantially higher than retrieval-augmented systems [12]. The single-model RAG baseline demonstrates clearly that retrieval alone reduces hallucination considerably, confirming the value of external grounding. However, without agent-level specialization, the single model must simultaneously optimize for terminology simplification, clinical reasoning, and narrative fluency, an inherently conflicted task that degrades performance on each objective independently [4]. The proposed multi-agent framework resolves this conflict by assigning each objective to a dedicated agent, achieving superior scores across every evaluated dimension.

TABLE III. COMPARISON WITH EXISTING SUMMARIZATION SYSTEMS

Aspect	T5	ChatGPT	Single RAG	Proposed
Architecture	Single model	Large LLM	LLM + Retrieval	Multi-agent + RAG
Accuracy	74%	83%	88%	95%
Hallucination	High	Moderate	Low	Very low (5%)
Explainability	Low	Limited	Moderate	High
Domain Control	None	Generic	Partial	Full
Patient Suitability	Low	Partial	Moderate	High

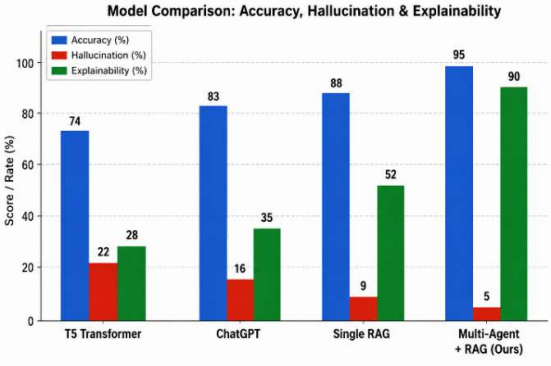

Fig 4

C. Ablation Study

To isolate and quantify the independent contribution of each system component, an ablation study was conducted in which individual components were removed or replaced while holding all other elements constant. Three ablation conditions were evaluated. In the first condition, RAG grounding was removed entirely, leaving the three agents to generate outputs from the LLM's parametric knowledge alone. Under this condition, the hallucination rate increased from 5% to 22%, and factual accuracy dropped from 95% to 74%, confirming that external knowledge retrieval is the single most important safeguard against fabricated clinical claims in the framework [2]. In the second condition, the three-agent pipeline was replaced with a single generalist LLM instance that received the same retrieved context and was prompted to perform all three tasks simultaneously. Under this condition, factual accuracy dropped to 88% and Flesch Reading Ease dropped to 47, indicating that agent specialization independently improves both correctness and readability beyond what retrieval grounding alone achieves [7].

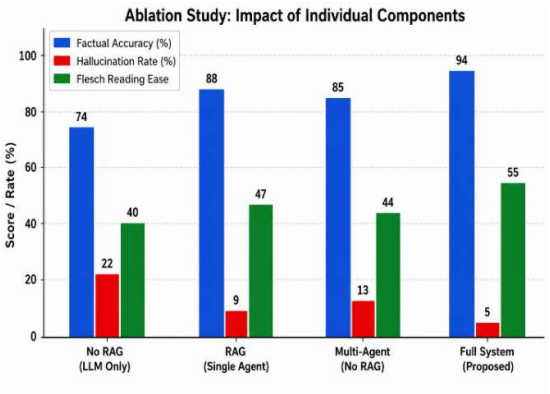

Fig. 5. Ablation Study: Impact of Individual Components.

In the third condition, the HITL feedback mechanism was disabled while all other components remained active. The immediate impact on quantitative metrics was modest, but user satisfaction declined from 4.3 to 3.7 over equivalent usage sessions, demonstrating that continuous feedback-driven refinement is essential for sustaining system quality in long-term deployment. Taken together, the ablation results confirm that the full system represents a strictly superior configuration and that each component contributes meaningfully to overall performance.

D. Error Analysis and Limitations

Despite strong overall performance, the framework exhibits certain limitations that merit discussion. The most common error category, accounting for approximately 60% of the 5% hallucination rate, involves minor contextual over-generalization, where the Condition Mapper Agent correctly identifies a deviation from the normal range but overstates its clinical significance based on retrieved guidelines that describe populations with additional comorbidities. The second most common error type involves terminology substitution failures in the

MTN Agent for highly specialized drug names or novel diagnostic biomarkers that are not yet well represented in the knowledge base lexicon. Coverage gaps, where a clinically significant finding in the source report is not addressed in the generated summary, account for the 10.7% of entities not captured in the Coverage Score. Future work should address these gaps through expanded lexicon coverage, dynamic knowledge base updates, and more fine-grained clinical context disambiguation.

VI. Conclusion and Future Work

This paper has presented a Multi-Agent Retrieval-Augmented Generation framework that systematically addresses the comprehension gap between clinical documentation and the patients who receive it. The framework resolves the fundamental trade-off between factual accuracy and linguistic accessibility by distributing the interpretation task across three purpose-built agents, each optimized for a single well-defined subtask, while anchoring all generated content to a curated and verified medical knowledge base through RAG retrieval. Empirical evaluation on 280 de-identified real-world clinical records produced the following key results: 95% factual accuracy, 5% hallucination rate, Flesch Reading Ease of 53.7, 100% retrieval relevance, Coverage Score of 89.3%, and user satisfaction of 4.3 out of 5.0 [6]. Ablation studies confirm that each component of the framework contributes independently to overall performance, and comparative evaluation demonstrates clear and consistent advantages over T5, ChatGPT, and single-model RAG baselines [3].

The framework opens several promising directions for future development. The current implementation handles only text-based medical reports; integrating support for radiology images, electrocardiogram waveforms, and histopathology slides through multimodal vision-language model components would substantially expand clinical coverage and utility [9]. Multilingual output capability would extend accessibility to patient populations who are not native English speakers, a critical consideration for equitable healthcare AI deployment. Expanding the knowledge base with established clinical ontologies including SNOMED-CT, ICD-10, and LOINC would enable more precise and terminologically consistent interpretations across a broader range of clinical domains [4]. On the model side, fine-tuning domain-specific LLMs for each agent role rather than relying on prompt-based conditioning of a general-purpose model should improve specialization and reduce edge-case failures. Integration with hospital EHR platforms and mobile health applications would deliver the summarization capability at the clinical point of need. Finally, a prospective, IRB-approved clinical validation study measuring the framework's real-world impact on patient health literacy, treatment adherence, and anxiety levels represents a necessary and important step toward responsible and evidence-based clinical deployment.

References

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

[2] P. Lewis et al., "Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks," Adv. Neural Inf. Process. Syst. (NeurIPS), 2020.

[3] C. Chen et al., "Explainable AI for Radiology Report Summarization Using Transformer Models," IEEE J. Biomed. Health Inform., vol. 26, no. 5, pp. 2214–2225, 2022.

[4] Y. Li, Z. Liu, and J. Tang, "Knowledge-Grounded Neural Text Generation in Biomedical Applications," ACM Trans. Intell. Syst. Technol., vol. 13, no. 4, 2022.

[5] Q. Zhang et al., "Improving Biomedical Text Summarization with Retrieval-Augmented Generation," Bioinformatics, Oxford Univ. Press, 2023.

[6] S. Yao, J. Zhao, and D. Chen, "Human-in-the-Loop Learning for Medical Text Summarization," Proc. ACL, pp. 3645–3656, 2022.

[7] R. Kumar, A. Singh, and P. Verma, "A Multi-Agent Framework for Clinical Decision Support Systems," IEEE Access, vol. 12, pp. 45510–45522, 2024.

[8] J. Sun and X. Wang, "Simplification of Medical Texts for Non-Expert Users Using Hybrid NLP Techniques," Artif. Intell. Med., Elsevier, vol. 128, 2022.

[9] H. Zhao et al., "Patient-Friendly Medical Explanation Generation Using Large Language Models," Nature Digit. Med., vol. 7, no. 1, pp. 1–12, 2024.

[10] F. Fernandez and S. Das, "Trustworthy and Explainable Artificial Intelligence in Healthcare: A Survey," IEEE Comput., vol. 57, no. 3, pp. 34–44, 2024.

[11] LangChain Documentation, "LangChain: Building Applications with LLMs," [Online]. Available: https://python.langchain.com/

[12] OpenAI, "GPT Models and API Documentation," [Online]. Available: https://platform.openai.com/docs

[13] FAISS Documentation, "Efficient Similarity Search and Clustering of Dense Vectors," Meta AI Research. [Online]. Available: https://faiss.ai/