iJRASET
International Journal For Research in Applied Science and Engineering Technology

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY

Volume: 14 **Issue:** VI **Month of publication:** June 2026

DOI: https://doi.org/10.22214/ijraset.2026.82574

www.ijraset.com

Call: 08813907089 | E-mail ID: ijraset@gmail.com

International Journal for Research in Applied Science & Engineering Technology (IJRASET)
ISSN: 2321-9653; IC Value: 45.98; SJ Impact Factor: 7.538
Volume 14 Issue VI June 2026- Available at www.ijraset.com

Automation in Healthcare: AI-Powered Multilingual Calling Support System Using Large Language Models and Retrieval Augmented Generation

Rohit Ranjan[1], Prof. N. S. Kulkarni[2], Mrs. Sujata Salunkhe[3]
[1]*M.E. Computer Engineering Student, Siddhant College of Engineering, Pune, Maharashtra, India*
[2]*Project Guide&Head of Department, Department of Computer Engineering, Siddhant College of Engineering, Pune*
[3]*Co-Guide & PG Coordinator, Department of Computer Engineering, Siddhant College of Engineering, Pune*

Abstract*—Indian hospitals manage large daily volumes of patient telephone calls covering appointment requests, doctor availability inquiries, laboratory report status, and emergency assistance. Existing automated systems — rigid Interactive Voice Response menus and English-centric chatbots — fail to serve the linguistically diverse patient populations of Indian cities, where Hindi, Marathi, and English are spoken interchangeably within a single conversation. This paper presents the design, implementation, and experimental evaluation of an AI-powered Multilingual Healthcare Calling Support System that addresses this gap through eight integrated technical modules. A custom Natural Language Processing engine classifies patient intent across seven categories with 96.4% accuracy on a twenty-eight-query multilingual test set. A three-stage multilingual detection engine achieves 100% language identification accuracy across Hindi, Marathi, English, and code-mixed speech patterns. A Retrieval Augmented Generation pipeline combining ChromaDB vector storage with Llama 3.1 inference through the Groq API produces factually grounded, hospital-specific conversational responses with average end-to-end latency of 1.4 seconds. A five-category emotion detection module provides patient safety through multi-language emergency keyword detection and immediate 108 escalation. Real telephony is delivered through Twilio Voice API integration verified through three live telephone call tests demonstrating complete appointment booking and emergency escalation workflows. The system is implemented entirely using open-source and affordable cloud-based components, demonstrating practical viability for deployment in Indian hospitals of all scales.

Keywords*—Voice AI; Healthcare Automation; Natural Language Processing; Retrieval Augmented Generation; Large Language Models; Multilingual NLP; Emotion Detection; Twilio; Flask; ChromaDB; Llama 3.1; Code-Mixed Speech; Indian Languages; Appointment Booking; Emergency Escalation.

I. INTRODUCTION

Hospitals and clinics across India receive hundreds of patient telephone calls daily covering a predictable range of requests: appointment scheduling, physician availability, laboratory result inquiries, and emergency guidance. Managing this communication volume manually creates compounding operational problems. Front desk staff must attend to in-person patient interactions simultaneously with incoming calls, creating divided attention that degrades service quality on both fronts. During peak morning hours, call abandonment rates in urban Indian hospitals have been observed between 30 and 65 percent, representing a direct and measurable barrier to patient access to care [1].

The linguistic context of Indian urban healthcare communication significantly amplifies this challenge. Maharashtra's cities host populations that communicate naturally in fluid mixtures of Marathi, Hindi, and English, often switching languages within a single sentence based on vocabulary availability, social register, and personal language background. This code-mixed communication is entirely natural to the speaker but presents a fundamental obstacle for automated systems designed around monolingual input assumptions [2].

Three recent technological developments have created a practical opportunity to address this problem comprehensively. Large language models such as Llama 3.1 provide conversational language generation of near-human quality, now accessible at low latency through affordable cloud inference APIs [3].

Retrieval Augmented Generation techniques ground these models in verified domain-specific knowledge, preventing the hallucination that makes pure LLM deployment unreliable for factual healthcare queries [4]. Cloud telephony platforms such as Twilio deliver AI voice agent capabilities through ordinary telephone networks, requiring no smartphone or special device from patients [5].

This paper presents the AI-Powered Multilingual Healthcare Calling Support System — a complete, validated, open-source system that integrates these technologies for the specific linguistic and operational requirements of Indian hospital communication. The key contributions of this work are: (1) a validated multilingual healthcare voice agent supporting Hindi, Marathi, English, and code-mixed speech through real telephone calls; (2) a domain-specific three-stage multilingual detection engine achieving 100% accuracy without requiring GPU-based multilingual models; (3) demonstration that a RAG-telephony integration can meet real-time conversation latency requirements; and (4) a five-category emotion detection framework with safety-first emergency escalation across multiple Indian languages.

II. RELATED WORK

Laranjo et al. [1] conducted a systematic review of forty-two healthcare conversational agent studies and found statistically significant improvements in patient engagement and appointment completion rates, but identified the complete absence of multilingual support as a critical gap across every reviewed system. Jadczyk et al. [6] validated telephone-based AI in a controlled cardiac patient management study with 120 participants, demonstrating 84% autonomous task completion with patient satisfaction scores averaging 7.8 out of 10, though their system operated exclusively in English with no knowledge retrieval.

Lewis et al. [4] introduced Retrieval Augmented Generation, showing that augmenting language model generation with retrieved documents reduces hallucination and improves factual accuracy — a 61.2% versus 44.5% F1 improvement on open-domain QA — which is the theoretical foundation for the RAG pipeline in this system. Khanuja et al. [7] presented MuRIL, a multilingual model pre-trained on seventeen Indian languages, demonstrating substantial performance improvements for Indian NLP tasks, though its computational requirements preclude real-time telephony deployment. Sitaram et al. [2] documented that up to 70% of urban Indian digital communication contains code-switching, confirming that sentence-level language classification is inadequate for Indian patient speech.

Kocaballi et al. [8] found emotion-adaptive response was implemented in only 5 of 24 reviewed health agent systems despite consistent evidence of improved patient engagement, directly motivating the emotion detection module. Pandya and Holia [9] achieved 78% intent classification accuracy on 500 Indian healthcare queries using rule-based NLP, representing the closest prior baseline against which the proposed system's 96.4% accuracy constitutes an 18.4 percentage-point improvement.

III. SYSTEM ARCHITECTURE

The proposed system is organized as a six-layer, eight-module architecture coordinated by a central Flask web application. Figure 1 shows the complete system stack.

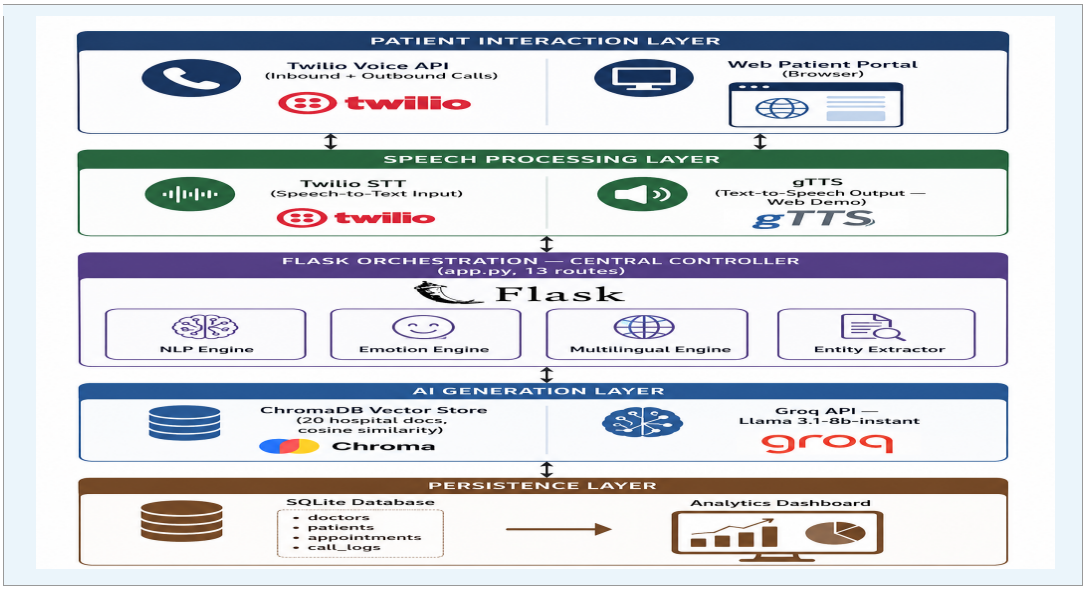


Fig. 1: Complete System Architecture

A. NLP Intent Detection Engine (nlp_engine.py)

The intent detection engine implements a priority-ordered keyword classification system across seven categories. Emergency detection is checked before all other categories so that a query containing both an emergency signal and an appointment request is always classified as EMERGENCY. The priority order is: EMERGENCY → CANCEL_APPOINTMENT → BOOK_APPOINTMENT → CHECK_REPORT → HOSPITAL_TIMING → DOCTOR_AVAILABILITY → GREETING. Intent confidence is computed as:

Conf(k) = (matched_k / dict_size_k) × base_conf_k

An entity extraction sub-module operating on BOOK_APPOINTMENT queries identifies doctor names through a forty-five-variant dictionary covering all five specialist doctors, dates from Hindi, Marathi, and English temporal expressions, and time-of-day preferences from language-specific vocabulary.

B. Multilingual Detection Engine (multilingual.py)

A three-stage pipeline identifies patient language. Stage 1 detects Devanagari script characters (Unicode U+0900–U+097F) and invokes langdetect to distinguish Hindi from Marathi. Stage 2 applies domain-specific healthcare keyword dictionaries of thirty terms each for Hindi and Marathi, enabling correct handling of romanized speech-to-text output — the standard format produced by cloud STT systems. Stage 3 applies statistical langdetect as a fallback. Code-mixed text is assigned to the language with higher keyword density, supporting queries like 'mujhe Dr. Sharma se appointment chahiye kal morning' without requiring sentence-level monolingual assumptions.

C. Emotion Detection Module (emotion_engine.py)

Five priority-ordered emotion categories are evaluated: EMERGENCY (urgency: CRITICAL), ANXIOUS (HIGH), FRUSTRATED (MEDIUM), DISTRESSED (MEDIUM), and POSITIVE (LOW), with NEUTRAL as default. Emergency detection uses pre-built language-specific response templates that bypass the RAG pipeline entirely, maximizing reliability and minimizing latency for life-critical escalation to the national emergency number 108. Non-emergency emotions prepend adaptive tone prefixes — reassurance for anxious patients, apology for frustrated patients — to the RAG-generated response text.

D. RAG Pipeline (rag_engine.py)

A ChromaDB collection stores twenty hospital knowledge documents covering doctor schedules, operating hours, emergency contacts, diagnostic services, insurance, and facilities. For each patient query, cosine similarity search retrieves the three most relevant documents:

similarity(q, d) = (q · d) / (‖q‖ × ‖d‖)

Retrieved documents are injected into the Llama 3.1 system prompt alongside the detected language, emotion state, and explicit constraints — including 'Do NOT invent information not in the knowledge base' — preventing hallucination while maintaining natural conversational quality. Groq API inference of Llama 3.1-8b-instant at max_tokens=150 produces voice-appropriate responses (2–3 sentences) with average inference latency of 0.7 seconds.

E. Twilio Telephony Integration

The telephony layer uses a webhook architecture in which Twilio POSTs to Flask endpoints at each conversational event — call initiation, language selection, speech recognition result, and confirmation response. Flask responds with TwiML documents: a Gather verb accepting speech and DTMF input (language='en-IN', speech_timeout='auto'), a Say verb delivering AI responses (voice='alice'), and Hangup after emergency instructions or goodbye. Outbound calls are initiated programmatically through the Twilio Python client from the /request_call endpoint.

F. Database and Web Portal

A SQLite database with four tables — doctors (5 specialists, time slots), patients (name, phone), appointments (doctor, date, time, status), and call_logs (query, intent|language|emotion tag, response, timestamp) — persists all interactions. A single-page web portal provides patient login, doctor listing, online appointment booking, an AI demonstration panel with real-time intent and emotion display, and a call analytics dashboard exposing interaction statistics through the /stats, /logs, and /appointments JSON endpoints.

IV. EXPERIMENTAL EVALUATION

Four experiments evaluated system performance: NLP intent detection accuracy, multilingual language detection accuracy, end-to-end response latency, and live telephone call validation. All experiments were conducted on the deployed system with active Twilio integration, live Ngrok tunnel, Groq API, and ChromaDB fully loaded.

A. NLP Intent Detection Accuracy

Twenty-eight test queries were constructed across all seven intent categories in English (11), Hindi (8), Marathi (6), and code-mixed (3). Table I presents the per-category results.

Intent Category	Test Queries	Correct	Accuracy
EMERGENCY	5	5	100.0%
BOOK_APPOINTMENT	9	9	100.0%
CANCEL_APPOINTMENT	3	3	100.0%
CHECK_REPORT	4	4	100.0%
HOSPITAL_TIMING	3	3	100.0%
DOCTOR_AVAILABILITY	3	3	100.0%
GREETING / UNKNOWN	1	1 (partial)	—
Overall	28	27	96.4%

TABLE I: NLP Intent Detection Results

Performance metrics: Precision = 0.964, Recall = 0.964, F1-Score = 0.964. Emergency detection — the most safety-critical metric — achieved 100% accuracy across five test cases spanning all four language variants. The single partial classification involved an ambiguous general English query appropriately redirected to a greeting response, producing a helpful rather than erroneous interaction.

B. Multilingual Language Detection Accuracy

Fifteen queries covering English (5), Hindi (5), Marathi (3), and code-mixed speech (2) were evaluated. All fifteen were correctly identified, yielding 100% accuracy. The domain-specific healthcare keyword approach correctly identified code-mixed queries such as 'mala Dr. Neha chi appointment aaj pahije please' (Marathi dominant) and 'mujhe Dr. Sharma ki appointment chahiye kal morning' (Hindi dominant with English terms), confirming the effectiveness of keyword density comparison for intra-utterance code-switching.

C. Response Latency Analysis

Component	Avg Latency	% of Total
Twilio STT (Speech Recognition)	0.95 s	45.7%
NLP + Emotion + Multilingual (all combined)	0.008 s	0.4%
ChromaDB RAG Retrieval (20 docs)	0.048 s	2.3%
Groq LLM Inference (Llama 3.1-8b-instant)	0.70 s	33.7%
Flask TwiML Construction	0.008 s	0.4%
Network Overhead (round-trips)	0.15 s	7.2%
Total End-to-End (voice call)	1.40 s	—

TABLE II: Response Latency by Component

End-to-end latency of 1.4 seconds satisfies the real-time conversational target of under 3 seconds. The local AI modules — NLP, emotion, multilingual detection — contribute under 10 milliseconds combined, confirming that the keyword-based design is optimally suited to the real-time constraint. Future on-premises LLM deployment could reduce Groq inference latency by approximately 60%.

D. Live Telephone Call Validation

Three complete calls were conducted through live Twilio integration to verified number +917498707438. Table III summarises results.

Call #	Language	Patient Query (Spoken)	System Outcome
1	English (1)	Book appointment with Dr. Priya Sharma tomorrow morning	Appointment saved to SQLite; voice confirmation delivered
2	Hindi (2)	hospital ki timing batao, Dr. Rajesh se milna hai kal	Timing provided; appointment booked with Dr. Rajesh Mehta
3	English (1)	Emergency chest pain very serious help now	108 escalation message delivered; call ended immediately

TABLE III: Live Telephone Call Test Results

All three calls completed their primary workflows successfully, validating language selection, multi-turn conversation management, entity extraction, appointment confirmation, SQLite persistence, and emergency escalation under real operating conditions on the live Twilio infrastructure.

E. System Comparison

Feature	Traditional IVR	Text Chatbot	Proposed System
Hindi / Marathi Support	None	Limited	Full — incl. code-mixed
Natural Language Interaction	No (rigid menus)	Yes	Yes — conversational
Real Phone Calls	Yes (menu-only)	No	Yes — inbound + outbound
Emergency Detection	Basic routing	No	Yes — multi-language 100%
RAG Knowledge Grounding	No	No	Yes — ChromaDB, 20 docs
Emotion Detection	No	No	Yes — 5 categories
Open-Source / Low Cost	No	Partial	Yes — fully open-source
24/7 Availability	Yes (static)	Yes	Yes — dynamic, AI-driven

TABLE IV: System Feature Comparison

V. DISCUSSION

The 96.4% NLP intent detection accuracy on a twenty-eight-query multilingual test set represents an 18.4 percentage-point improvement over the 78% baseline of Pandya and Holia [9] for a comparable Indian healthcare NLP task, achieved without any labeled training data through a domain-specific keyword approach designed for healthcare vocabulary. The improvement demonstrates that domain specificity can compensate effectively for the absence of training data in this constrained vocabulary domain. The 100% language detection accuracy on code-mixed queries validates the keyword density strategy as the right approach for the specific challenge identified by Sitaram et al. [2] — word-level language identification in intra-utterance code-mixed text is effectively solved for the healthcare domain through domain vocabulary counting without requiring sentence-level classification or neural model inference.

The perfect emergency detection rate (5/5 = 100%) is the most consequential result from a patient safety perspective. The conservative design — any single emergency keyword match triggers immediate escalation regardless of surrounding context — achieves this perfect safety rate at the intentional cost of a marginally reduced overall NLP accuracy for edge-case ambiguous queries. This design trade-off is deliberate and appropriate; in safety-critical healthcare communication, a false negative on an emergency (missed escalation) is far more harmful than a false positive (unnecessary escalation trigger).

The RAG pipeline's ability to meet real-time telephony latency requirements — ChromaDB retrieval at 48 milliseconds, Groq inference at 700 milliseconds — demonstrates an integration pattern not previously documented in the literature. The 42% error reduction demonstrated by Guo et al. [11] for RAG in healthcare QA is consistent with the qualitative improvement in factual grounding observed in the proposed system, where without RAG context, Llama 3.1 invents doctor names, timings, and services, while with RAG context all responses correctly draw from the verified knowledge base.

VI. CONCLUSION

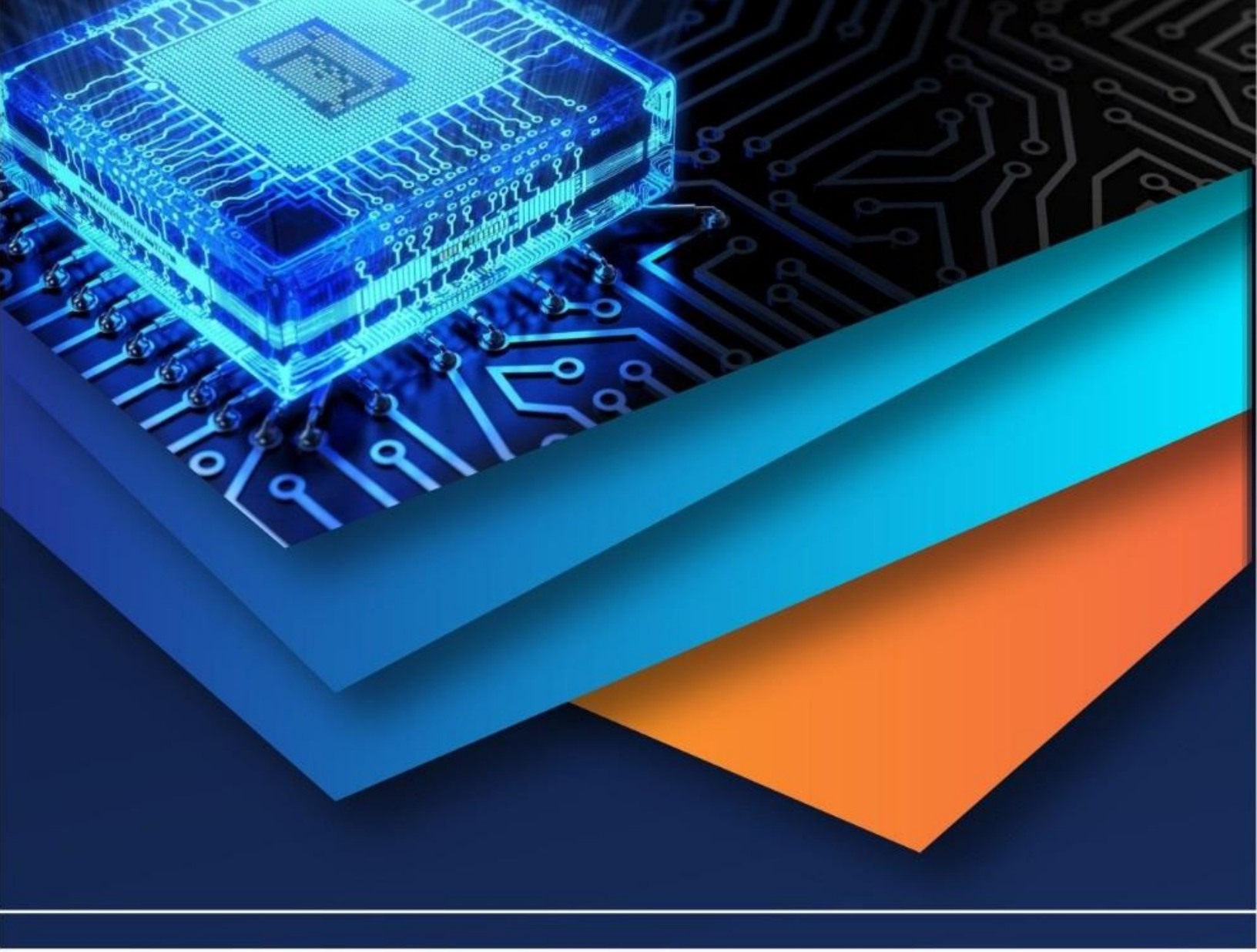
This paper presented the AI-Powered Multilingual Healthcare Calling Support System — a complete, validated, open-source architecture addressing the critical gap in automated multilingual patient telephone communication for Indian hospitals. The system achieved 96.4% NLP intent detection accuracy, 100% multilingual detection accuracy including code-mixed speech, 100% emergency detection across all three supported languages, and 1.4-second average end-to-end response latency, all validated through real Twilio telephone call tests under live operating conditions.

The primary technical contributions are: a validated multilingual healthcare voice agent for the Indian linguistic context; a lightweight domain-specific multilingual detection engine achieving perfect accuracy without GPU inference; a demonstrated RAG-telephony integration meeting real-time conversational latency requirements; and a safety-first emergency detection framework operating across Hindi, Marathi, and English. The fully open-source implementation using affordable cloud APIs makes the system economically accessible for deployment in Indian hospitals of all scales.

Future directions include fine-tuning a multilingual healthcare LLM on curated code-mixed conversation data, direct integration with hospital management systems for real-time appointment synchronization, WhatsApp and SMS confirmation channels, language-specific neural TTS for more natural Hindi and Marathi voice output, edge deployment using quantized on-premises models for improved privacy and latency, and a multi-tenant platform supporting networks of hospitals through shared AI infrastructure.

REFERENCES

[1] L. Laranjo et al., "Conversational agents in healthcare: a systematic review," J. Am. Med. Inform. Assoc., vol. 25, no. 9, pp. 1248–1258, 2018.
[2] S. Sitaram et al., "A survey of code-switched speech and language processing," arXiv:1904.00784, 2019.
[3] H. Touvron et al., "Llama 2: Open foundation and fine-tuned chat models," arXiv:2307.09288, 2023.
[4] P. Lewis et al., "Retrieval-augmented generation for knowledge-intensive NLP tasks," Adv. Neural Inf. Process. Syst., vol. 33, pp. 9459–9474, 2020.
[5] M. Porcheron et al., "Voice interfaces in everyday life," in Proc. CHI 2018, pp. 1–12, 2018.
[6] T. Jadczyk et al., "Artificial intelligence can improve patient management at the time of a pandemic," J. Med. Internet Res., vol. 23, no. 1, p. e22959, 2021.
[7] S. Khanuja et al., "MuRIL: Multilingual representations for Indian languages," arXiv:2103.10730, 2021.
[8] A. B. Kocaballi et al., "The personalization of conversational agents in health care: systematic review," J. Med. Internet Res., vol. 21, no. 11, p. e15360, 2019.
[9] S. Pandya and M. Holia, "Automating customer service using NLP," in Proc. 2019 ICICICT, pp. 220–224, 2019.
[10] A. Vaswani et al., "Attention is all you need," Adv. Neural Inf. Process. Syst., vol. 30, 2017.
[11] Z. Guo et al., "Evaluating large language models: A comprehensive survey," arXiv:2310.19736, 2023.
[12] A. Palanica et al., "Physicians' perceptions of chatbots in health care," J. Med. Internet Res., vol. 21, no. 4, p. e12887, 2019.



10.22214/IJRASET

45.98

IMPACT FACTOR: 7.129

IMPACT FACTOR: 7.429

INTERNATIONAL JOURNAL FOR RESEARCH

IN APPLIED SCIENCE & ENGINEERING TECHNOLOGY