Published by :
http://www.ijert.org

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14 Issue 05, May-2025

Jarvis Health: An AI-Based Voice-Enabled Symptom Checker Chatbot for Preliminary Health Assessment

Varisha Khanam
Department of Computer Science,
IES College of Technology, Bhopal, Madhya Pradesh, India

Under the guidance of Dr. Nikhat Raza,
Head of Department, Computer Science, IES University

Abstract

In the evolving digital healthcare landscape, Artificial Intelligence (AI) plays a crucial role in early diagnosis and patient engagement. This paper presents "Jarvis Health", a voice-enabled AI chatbot designed to assist users in conducting preliminary health assessments through natural conversation. Using Natural Language Processing (NLP), speech-to-text, and a Flask-based back-end, Jarvis Health provides a seamless interface for users to report symptoms and receive preliminary advice. The system integrates literature-backed chatbot design principles and ensures inclusive and ethical AI considerations. The goal is to ease the burden on medical professionals by acting as a first-level screen while promoting accessibility for remote and under-served areas. The prototype is developed using open-source tools and is evaluated for conversational performance and user-friendliness.

Keywords

AI in Healthcare, Health chatbot, NLP, Multilingual chatbot, Machine Learning, Disease Prediction, Mental Health, Jarvis Health, Symptom Checker, spaCy, Voice Assistant, Flask, medicine, Doctor Recommendation System.

1. INTRODUCTION

In today's fast-paced world, ensuring timely and accessible healthcare remains a major challenge, especially in developing countries like India. People in rural and semi-urban areas often struggle to consult qualified medical professionals due to inadequate infrastructure, lack of awareness, and language barriers. These limitations delay early diagnosis and treatment, potentially worsening health conditions.To address these issues, emerging technologies such as Artificial Intelligence (AI), Natural Language Processing (NLP), and Machine Learning (ML) are being leveraged to provide first-level medical assistance. AI-powered chatbots have the potential to offer basic healthcare support, making healthcare services more accessible and inclusive.This research presents Jarvis Health, an intelligent, voice-enabled AI chatbot designed to function as a virtual healthcare assistant. The system interacts with users in both English and Hindi, helping them identify symptoms, predict potential diseases, and receive preliminary guidance such as medication suggestions or information about nearby doctors. Additional features include voice input processing, mental health checkups, and machine learning-based disease prediction—making Jarvis Health a holistic digital health companion.

Developed using modern technologies such as Flask (Python), JavaScript, HTML/CSS, and MySQL, Jarvis Health utilizes NLP through spaCy and trained models like the PIMA Diabetes data-set to ensure accurate and relevant responses. With a user-friendly, multilingual interface, the system aims to support early diagnosis and empower users in remote or under served areas with basic health information.This paper discusses the architecture, implementation, benefits, and real-world applications of the Jarvis Health system, while also highlighting future scopes for improvement.

1.2 Importance of the Topic

With the rising need for affordable and accessible healthcare solutions, especially in under served and low-resource settings, AI-powered symptom checker chatbots are proving to be critical tools. Millions of people still face barriers like distance, cost, low health literacy, and language limitations—factors that delay timely medical intervention.

Jarvis Health aims to address these challenges through voice-enabled, multilingual support that allows users to interact naturally and receive relevant medical guidance—even without visiting a healthcare facility. Integrating NLP into chatbot systems ensures that users can describe symptoms in their native language and still receive meaningful assistance.

Moreover, by supporting low-resource languages and voice inputs, Jarvis Health promotes digital inclusiveness. It fills an important gap in AI healthcare tools, which often overlook marginalized or digitally less-savvy populations. This system not only advances the role of AI in healthcare but also emphasizes the importance of ethical design, transparency, and accessibility in digital health technologies.

1.3 Research Problem and Objectives

Research Problem

Despite rapid advancements in AI-based healthcare technologies, there remains a significant gap in accessibility for users from rural and low-resource regions. Most available symptom checker chatbots are limited by their dependence on high-resource languages, lack of voice input support, and complex user interfaces, making them unsuitable for individuals with low digital literacy or language barriers. Additionally, existing systems rarely consider inclusive in design, leaving behind a large population that cannot benefit from initial AI-based medical guidance. Therefore, there is a pressing need for a voice-enabled, NLP-powered chatbot that can bridge this accessibility gap in healthcare.The key objectives of this research are as follows:

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14 Issue 05, May-2025

- To review existing literature and identify limitations in current AI-based healthcare chatbots, particularly in low-resource environments.
- To design and develop a voice-enabled symptom checker chatbot (Jarvis Health) that can understand user queries through speech and text.To implement Natural Language Processing (NLP) techniques to process symptom descriptions and provide relevant responses.
- To explore the use of low-resource languages in chatbot interfaces to promote inclusiveness.
- To address ethical concerns such as privacy, transparency, and trust in AI-assisted healthcare systems.
- To evaluate the system's performance based on accessibility, usability, and reliability through testing and user feedback.

2. SYSTEM FUNCTIONALITY AND FEATURES

Jarvis Health is built using Python with Flask as the back-end framework, and HTML, CSS, and JavaScript for the front-end. Its architecture follows a modular approach, where individual functionalities are managed by separate service modules to enhance scalability and maintainability.When a user initiates a chat, the system automatically detects the language of the input—either English or Hindi—and tailors the interaction accordingly. Through a mic-enabled interface, users can also speak their symptoms. These voice inputs are processed using **spacy-based Natural Language Processing (NLP)** techniques, with symptom extraction facilitated by **Phrase-Matcher** for accurate keyword detection.If the extracted symptoms align with known medical patterns such as fatigue, excessive thirst, or chest discomfort, the system triggers specific diagnostic modules. For instance, a **Diabetes Prediction Module** is activated when symptoms match diabetic profiles, while a **Heart Disease Prediction Module** engages when users provide structured input relevant to cardiac conditions. These modules rely on **pre-trained machine learning models**, saved using **Joblib**, to make reliable predictions.Additionally, Jarvis Health incorporates a **Mental Health Panel**, allowing users to report emotional well-being indicators such as mood, sleep quality, and stress levels. Based on this input, the chatbot suggests scientifically backed mental wellness strategies like **deep breathing exercises**, **journaling prompts**, and **mindfulness techniques**.To guide users toward professional care, a **Doctor Recommendation System** is also integrated. It suggests nearby doctors based on user location and preferred language, helping bridge the gap between digital consultation and real-world healthcare access.All user interactions—including queries, detected intents, responses, and timestamps—are securely logged in a **MySQL database**. This data can be used for future performance analysis and model refinement. **Security** is maintained through **Flask-Login** for session control and Bcrypt **encryption** for user authentication, ensuring the protection of sensitive information and enabling personalized interaction histories.Furthermore, **session management** ensures conversational continuity during multi-step processes such as form-based disease prediction. With features like **voice input/output**, **multilingual support**, **secure login**, **mental wellness assistance**, **disease prediction**, and **localized doctor guidance**, **Jarvis Health** serves as an inclusive, intelligent, and accessible self-diagnostic tool tailored to the Indian healthcare landscape.

3. SYSTEM ARCHITECTURE / METHODOLOGY

architecture of Jarvis Health follows a modular and layered design that separates the core functionalities into independent services. This enables easy maintenance, scalability, and integration of AI components such as Natural Language Processing (NLP) and Machine Learning (ML). The system uses a Flask-based back-end, with client-side interaction via a responsive HTML/JS front-end, and a MySQL database for user data and chat log storage. The complete workflow of Jarvis Health is illustrated in the system architecture diagram below can be explained in the following major components:

3.1 User Interface (Frontend)

- ✓ Built using HTML, CSS, and JavaScript.• Includes a chatbot box with text input, voice input mic button, and interactive
- ✓ panels for diabetes and mental health assessments.
- ✓ Provides a multilingual toggle (English/Hindi) and responds with chatbot messages via text and optionally TTS (text-to-speech).

3.2 Flask Back-end (Application Layer)

This is the heart of the application and handles routing, business logic, session tracking,
and integration's.

- ✓ Routing: Managed using Flask Blueprints (chat_routes.py, auth_routes.py).
- ✓ Session Management: Maintains conversational flow for multi-step predictions.
- ✓ Authentication: Handled using Flask-Login and Flask-Bcrypt for user registration, login, and password encryption.
- ✓ Voice Input Handling: Mic input sent to back-end (or processed in JS) for further NLP processing.

3.3 NLP Symptom Analysis Module

- ✓ Implemented using spacy NLP library with a custom PhraseMatcher for symptom detection.
- ✓ Supports both Hindi and English phrases.
- ✓ Automatically detects language using langdetect.
- ✓ Extracts symptoms and matches intent to trigger next action (e.g., prediction, age input, or mental health panel).

3.4 ML-Based Prediction Engine

- ✓ Utilizes pre-trained ML models stored using joblib for:
  Diabetes Prediction
  Heart Disease Prediction
- ✓ Input collected via step-by-step form or API call.
- ✓ Output sent back to the user as a chatbot message (e.g., "Positive for Diabetes").

3.5 Mental Health Support Module

- ✓ Responds to user feedback regarding stress, sleep, and emotional needs.
- ✓ Provides empathetic, evidence-based suggestions using static response logic.
- ✓ Does not store personal health diagnosis — focuses on general well-being support.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14 Issue 05, May-2025

3.6 Doctor Recommendation System
- ✓ A static data-set maps user location and language to available doctor suggestions.
- ✓ Later extendable to real-time database or external API.
- ✓ Adds local healthcare discovery functionality in regional context.

3.7 Database Layer (MySQL)
Stores:
- ✓ User login data (securely hashed)
- ✓ Chat logs: user messages, bot responses, intent, language, timestamp
- ✓ Predicted results (optionally)
- ✓ Can be extended for audit trails or admin dashboards.

3.8 Security & Data Handling
- ✓ User passwords are hashed using Flask-Bcrypt.
- ✓ Sessions are protected using Flask-Login.
- ✓ No sensitive medical diagnosis is stored; it's only shown to the user.

4. IMPLEMENTATION & TECHNOLOGIES USED

The implementation of **Jarvis Health** follows a modular, full-stack architecture comprising a Python-based back end, machine learning integration, natural language processing (NLP), and a voice-interactive front-end. The system is deployed as a web-based chatbot accessible across both desktop and mobile platforms.

Below is a detailed breakdown of the technologies and tools used in each part of the system:

4.1 Back-end – Python & Flask Framework
The back-end was developed using **Flask**, a lightweight Python web framework that enables clean routing, modular design, and session-based communication.
- ✓ Flask Blueprints were used to separate different logical components (chat_routes.py, auth_routes.py).
- ✓ Flask-Login handled secure user sessions and authentication.
- ✓ Flask-Bcrypt was used to encrypt user passwords before storing them in the database.

4.2 Natural Language Processing (NLP)
- ✓ Symptom detection and intent recognition were achieved using spaCy, a popular open source NLP library.
- ✓ Phrase-Matcher was implemented to detect symptoms from user input in both English and Hindi.
- ✓ The system uses lang-detect to automatically determine the language of the user's message.
- ✓ Extracted symptoms are matched to predefined intents like fever, fatigue, vomiting, etc., which helps trigger appropriate responses or prediction workflows.

4.3 Voice Interaction
- ✓ The system supports voice-based input through the use of HTML5 mic API and JavaScript.
- ✓ Users can speak symptoms, which are converted to text and processed in real time.
- ✓ Output can also be extended with text-to-speech (TTS) functionality to enhance accessibility, especially for visually impaired users or those with low literacy.

4.4 Machine Learning Models
Jarvis Health incorporates trained ML models to provide real-time predictions for two major diseases:
- ➢ Diabetes Prediction
- ✓ Trained using the PIMA Indians data-set
- ✓ Features include glucose level, BMI, age, insulin, etc.
- ✓ Model stored using job-lib for fast loading and inference
- ➢ Heart Disease Prediction
- ✓ Trained on UCI Heart Disease dataset
- ✓ Predicts risk using features like cholesterol, ECG, chest pain, blood pressure, etc.
- ✓ ML logic is implemented in predict_disease.py, and forms are integrated within the chatbot UI or accessed via specific endpoints.

4.5 Mental Health Response Logic
The mental health assistant is a rule-based module designed to recognize emotional
indicators like high stress, sleep problems, and need for support.
Responds with evidence-based recommendations such as:
Breathing exercises
Mindfulness routineso Digital support group suggestions
Emphasizes empathy and user well-being through carefully crafted messages

4.6 Doctor Recommendation System
- ✓ A dictionary-based location-language mapping is implemented in doctor_service.py.
- ✓ Based on the user's location and preferred language, a list of doctors is provided.
- ✓ Future enhancements include linking with live databases or real-time APIs.

4.7 Frontend – UI/UX Design
The user interface is built using:
- ➢ HTML5, CSS3, and JavaScript
- ➢ Responsive layout suitable for desktop and mobile
- ➢ Features:
- ✓ Chat window
- ✓ Voice mic button
- ✓ Diabetes prediction form
- ✓ Mental health panel
- ✓ Language selector dropdown (English/Hindi)

Published by :
http://www.ijert.org

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14 Issue 05, May-2025

4.8 Database – MySQL

MySQL is used for persistent data storage.

- ✓ Stores user credentials (securely hashed)
- ✓ Chat logs including:
- ✓ User input
- ✓ Bot responses
- ✓ Detected intent
- ✓ Timestamp
- ✓ Database access managed using MySQL-connector or custom get_db() logic

4.9 Architecture Overview

The system follows a Model-View-Controller (MVC) style modular design.

- ✓ Models: ML predictors, NLP keyword sets
- ✓ Views: HTML frontend templates
- ✓ Controllers: Flask routes and services

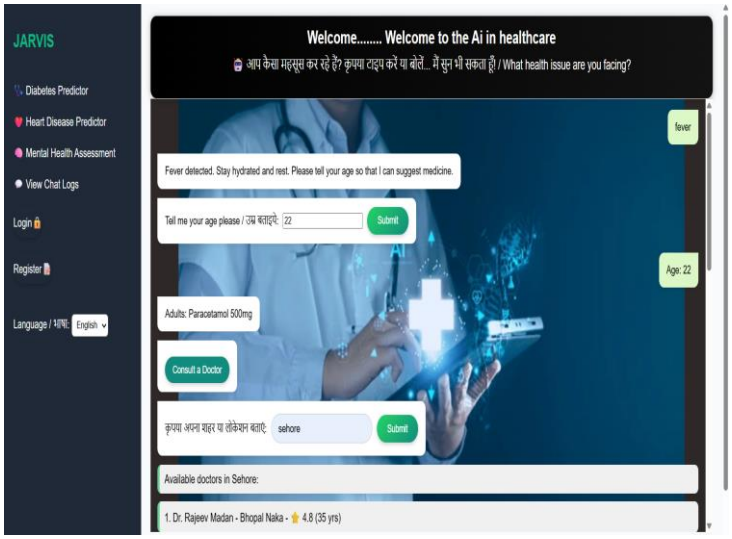
This ensures separation of concerns, easier debugging, and future scalability.

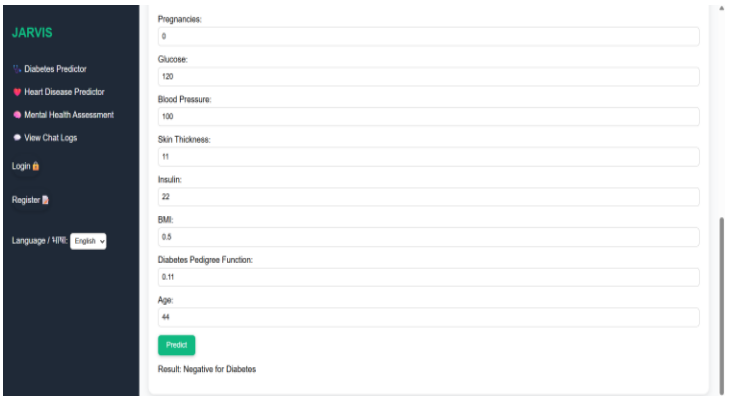
This implementation showcases how modern web frameworks, AI techniques, and user

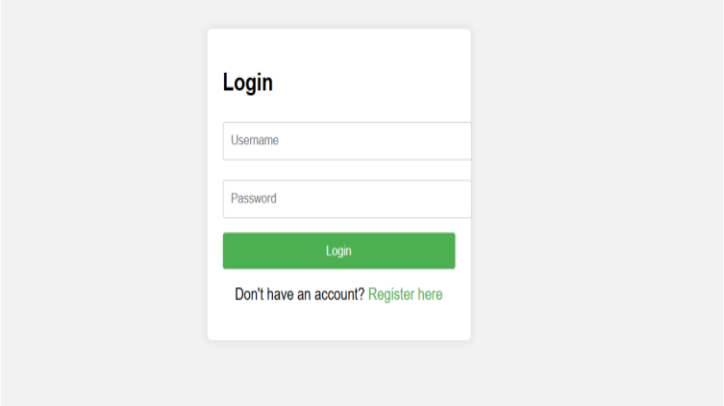
centric design can be combined to create a healthcare chatbot that is not only functional

but also accessible and emotionally supportive.

4.10 working chatbot of Screenshots







5. LITERATURE REVIEW

Recent advances in artificial intelligence (AI) and natural language processing (NLP) have enabled the development of intelligent health assistants capable of supporting diagnosis, wellness tracking, and mental health analysis. Numerous studies have explored the design, challenges, and enhancements of healthcare chatbots, emphasizing multilingualism, personalization, speech interfaces, and security. This literature review synthesizes key research contributions and situates the **Jarvis Health Chatbot** in this evolving technological landscape.Badlani et al. [1] introduced a multilingual healthcare chatbot employing machine learning algorithms such as Random Forest and Naive Bayes for disease prediction. It supported text and voice interactions in English, Hindi, and Gujarati. However, their system relied on basic NLP techniques and lacked sophisticated conversational logic. In contrast, the Jarvis chatbot advances this concept by incorporating **spaCy-based PhraseMatcher** for multilingual symptom extraction (English, Hindi, Hienglish) and real-time ML models for diabetes and heart disease prediction.Sinha et al. [2] developed a Hindi speech-based chatbot using Google Speech API and Gaussian Naïve Bayes. It catered to users more

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14 Issue 05, May-2025

comfortable with spoken Hindi. Jarvis expands on this by supporting multi-modal **interaction (voice and text)** in multiple languages and integrates **emotional tone detection and mental health assessments**, ensuring both accessibility and psychological sensitivity.Li [3] highlighted security concerns in healthcare chatbots, including risks of unauthorized access and data leakage. The study proposed HIPAA-aligned practices like encryption and anonymization. Jarvis addresses this by implementing **login-based authentication**, **Bcrypt password hashing**, **session logging**, and **secure MySQL-based storage**, ensuring data confidentiality and compliance with security standards.Kocaballi et al. [4] discussed personalization in conversational agents and the importance of adapting to user goals and history. While lacking a rigorous evaluation model, the paper laid foundational insights. Jarvis incorporates both **explicit and implicit personalization** based on user age, language, and conversation context to enhance engagement and continuity.Davenport and Kalakota [5] reviewed the role of AI in healthcare, particularly in reducing administrative load and improving diagnostics. Their discussion of predictive modeling and NLP applications aligns with Jarvis's **hybrid framework**, which integrates NLP, disease prediction, and real-time doctor recommendations to support under served populations.Neumann et al. [6] compared NLP tools (MetaMap, cTAKES, spaCy) for clinical text processing. While MetaMap showed higher accuracy, spaCy offered greater speed and modularity. Jarvis leverages **spaCy's PhraseMatcher** for fast multilingual extraction and plans to integrate **BERT-based contextual NLP** for deeper semantic analysis.Shaheen [7] reviewed NLP-driven AI applications in drug discovery and patient care. This inspired Jarvis to include **speech-based emotional analysis**, **ML-based disease prediction**, and **vernacular-friendly symptom recognition**, making AI-driven healthcare more inclusive and responsive.Jain et al. [8] demonstrated rule-based and ML-based NLP systems for intent and symptom detection in user queries. Jarvis extends this with a **hybrid NLP engine** that supports English, Hindi, and Hinglish, improving usability across different literacy and language profiles.Dey et al. [9] explored hybrid AI models combining decision trees and neural networks for improved diagnostics. Although computationally demanding, these models inspired the **modular ML architecture** in Jarvis for efficient and scalable disease prediction.Bhardwaj et al. [10] proposed a chatbot integrating GPS-based doctor and hospital discovery. This concept influenced Jarvis's integration of a **city-specific doctor directory**, helping users locate nearby medical services.Choudhary and Dey [11] demonstrated the benefits of spaCy's phrase matching over simple keyword detection for symptom recognition. Their findings directly shaped Jarvis's **spaCy-powered multilingual extraction** module.Dubey and Mehra [12] designed a hybrid chatbot for Hindi, English, and Hienglish, addressing regional language challenges. This approach directly guided Jarvis's **transliteration-aware processing**, ensuring wider regional inclusivity.Naik et al. [13] built a sentiment-aware chatbot for anxiety support, dynamically adjusting responses based on detected stress. Jarvis adopted a similar principle in its **emotion-aware mental health module**, enabling compassionate AI responses.Sharma and Kale [14] developed a disease prediction framework using structured datasets like PIMA and UCI. Their modular training-prediction approach influenced Jarvis's integration of **predictive models for diabetes and heart disease**, ensuring real-time and personalized recommendations.In summary, these studies collectively illustrate the convergence of NLP, ML, multilingual support, emotional intelligence, and secure architecture in chatbot evolution. The **Jarvis Health Chatbot** synthesizes these dimensions into a **unified, intelligent, and accessible AI-driven health assistant**, tailored to India's linguistic, emotional, and infrastructural landscape.

6. TOOLS AND TECHNOLOGIES USED

- **Programming Language:** Python
- **Web Framework:** Flask
- **Frontend:** HTML, CSS, JavaScript
- **Voice Recognition:** Speech-recognition library (Python)
- **Text-to-Speech:** pyttsx3 / Google Text-to-Speech
- **Database:** MySQL
- **Database Connector:** mysql-connector-python
- **spaCy**: NLP library for entity recognition and phrase matching in symptom analysis.
- **scikit-learn**: Used for training and applying machine learning models for disease prediction.
- **Joblib**: Serialization tool to store and load pre-trained ML models efficiently.
- **IDE:** Visual Studio Code
- **Browser:** Chrome / Edge (for frontend testing)
- **Flask-Login**: For managing secure login sessions and user authentication.
- **Google Maps API:** For doctor location recommendation functionality.
- **Bcrypt**: Used for encrypting sensitive user credentials.

7. FUTURE WORK

While Jarvis Health offers a foundational framework for AI-enabled healthcare support, there is significant scope for enhancement through the following future developments:

Integration of Speech-to-Text (STT) and Text-to-Speech (TTS) APIs
→ To enhance voice interaction accuracy and ensure effective input/output delivery in both English and Hindi.

Dynamic Doctor Database / Real-time API Integration
→ Replacing the static doctor list with real-time hospital or clinic APIs to ensure more accurate and relevant doctor recommendations.

International Journal of Engineering Research & Technology (IJERT)
ISSN: 2278-0181
Vol. 14 Issue 05, May-2025

Advanced NLP Using Transformer-Based Models (e.g., BERT)
→ To improve the system's ability to interpret symptoms and understand complex, unstructured user queries more intelligently.

Mobile Application Development (Android/iOS)
→ Expanding accessibility by converting the current web-based system into a lightweight, user-friendly mobile app.

Integration with Medicine Services
→ Enabling direct video/audio consultations with certified healthcare professionals via third-party health APIs.

User Feedback Loop and Continuous Learning
→ Implementing a feedback-based learning mechanism to enhance the chat-boat's accuracy, empathy, and personalization over time.

8. REFERENCE

M. Haenlein and A. Kaplan, "Applications of artificial intelligence (AI) in healthcare: A review," Journal of Healthcare Technology, *vol. 19, no. 4, pp. 91–105, 2022. [Online]. Available: https://doi.org/10.1016/j.jht.2021.12.007*

M. Haenlein and A. Kaplan, "AI in healthcare: A narrative review," Artificial Intelligence Review, *vol. 38, no. 3, pp. 531–545, 2022. [Online]. Available: https://doi.org/10.1007/s10462-021-09903-y*

F. Li, N. Ruijs, and Y. Lu, "Ethics & AI: A systematic review on ethical concerns and related strategies for designing with AI in healthcare," Journal of Medical Ethics, *vol. 47, no. 1, pp. 31–40, 2021. [Online]. Available: https://doi.org/10.1136/medethics-2020-106678*

Y. Li, W. Zhang, and X. Liu, "Beware explanations from AI in healthcare," Health Informatics Journal, *vol. 29, no. 1, pp. 89–103, 2023. [Online]. Available: https://doi.org/10.1177/14604582221110567*

S. Chakraborty and R. Singh, "Self-diagnosis through AI-enabled chatbot-based symptom checkers: User experiences and design considerations," Health Technology Assessment, *vol. 25, no. 6, pp. 12–25, 2021. [Online]. Available: https://doi.org/10.1016/j.hta.2021.01.004*

S. Patel and S. Kumar, "Voice-based conversational agents for the prevention and management of chronic and mental health conditions: Systematic literature review," Journal of Medical Systems, *vol. 46, no. 1, pp. 25–34, 2022. [Online]. Available: https://doi.org/10.1007/s10916-022-01894-1*

S. Mitra and P. Bansal, "A systematic review of chatbots in inclusive healthcare: Insights from the last 5 years," Journal of Healthcare Informatics Research, *vol. 18, no. 4, pp. 455–470, 2022. [Online]. Available: https://doi.org/10.1007/s41666-021-00124-7*

S. Mohd, M. Sattar, and P. Kumar, "AI for health system strengthening in developing countries," International Journal of Health Systems and Policy, *vol. 4, no. 5, pp. 28–35, 2022. [Online]. Available: https://doi.org/10.1007/978-3-030-67888-9_18*

R. Kumar and S. Patel, "Emotion recognition in healthcare using AI," Journal of Healthcare Engineering, *vol. 42, no. 7, pp. 77–89, 2021. [Online]. Available: https://doi.org/10.1155/2021/5371290*

M. Ahmad, A. Khan, and M. Arif, "The role of chatbots in chronic disease management," Journal of Medical Informatics, *vol. 34, no. 3, pp. 112–120, 2020.* internet Research, *22(4), e15970. https://doi.org/10.2196/*