**International Journal of Computer Sciences and Engineering**
Vol.**11**, Issue.**8**, pp.**56-64**, August **2023**
ISSN: 2347-2693 (Online)
Available online at: www.ijcseonline.org

ISSN: 2347-2693 (E)

Research Paper

# Medical Chatbot Using Sequence Modelling in Machine Learning

**S. Nithish Kumar[1*], S.Sujatha[2]**

[1,2]Dept. of Computer Applications, University College of Engineering, Anna University, BIT Campus, Tiruchirappalli, India

**Corresponding Author: nk193934@gmail.com*

**Received**: 07/Jul/2023; **Accepted**: 10/Aug/2023; **Published**: 31/Aug/2023. **DOI**: https://doi.org/10.26438/ijcse/v11i8.5664

**Abstract:** This project is focused on building a sophisticated chatbot that makes use of cutting-edge Natural Language Processing (NLP) methods to create seamless communication between users and healthcare practitioners. The main objective is to close the communication gap between medical professionals and those looking for quick answers to their health-related questions. The chatbot effectively understands user inputs by analyzing complex language correlations contained in their inquiries by utilizing NLP. Beyond solving the present drawbacks in remote healthcare encounters, this cutting-edge chatbot demonstrates the possibility for predictive diagnosis by spotting patterns in the symptoms that are regularly reported. This idea greatly improves the quality of remote medical consultations by fusing cutting-edge technology with healthcare. Improved patient care and outcomes are made possible by the prompt and accurate responses that are provided.

**Keywords:** Chat bot, Health care, Natural language processing (NLP), Medical expertise.

## 1. Introduction

The simulation of human intelligence processes by machines, particularly computer systems, is known as artificial intelligence (AI). These include reasoning—using rules to arrive at approximations or firm conclusions—learning—acquiring knowledge and rules for utilizing it—and self-correction. Machine vision, speech recognition, and expert systems are some specific uses of AI. There are other ways to classify AI, but here are two. An AI system that is created and taught for a specific task is referred to as weak AI, also known as narrow AI. Sometimes AI are referred to as artificial general intelligence, is an AI system that has human-like cognitive abilities that enable it to solve problems when faced with challenging tasks. Although the approach is debatable, the Turing Test was created by mathematician Alan Turing in 1950 to see if a machine can think like a human.

While AI tools offer a variety of new functions for enterprises, they also raise certain moral concerns. The deep learning algorithms that power many of the most cutting-edge AI products only have access to the training data. Most training data sets are likely to have some signs of bias. As a result, As a result, the AI technologies may exhibit bias in their performance. Ethical issues are brought up using AI in the field of self-driving cars.

Liability for accidents involving autonomous vehicles is not always obvious. In cases where an accident is unavoidable, autonomous cars may also be forced to make moral choices regarding how to limit the harm. The possibility for AI tools to be misused is yet another serious worry. Security is becoming more complicated than it already is as hackers begin to employ powerful machine learning technologies to access sensitive systems.

The study of building computer systems that can perceive their environment, interpret data, and make decisions in order to accomplish certain goals is known as artificial intelligence (AI). It attempts to imitate in robots intelligence and problem-solving skills similar to those of humans. AI makes it possible for computers to examine data, recognize patterns in it, and adjust to changing circumstances, ultimately enabling them to carry out cognitively demanding activities. A chatbot is a type of AI that uses software to have natural language conversations with users. It uses artificial intelligence (AI) methods to comprehend user inputs, produce suitable answers, and mimic human-like interaction. Chatbots can be used for a variety of tasks, from responding to consumer questions to making tailored recommendations. Chatbots increase user engagement, expedite interactions, and provide effective answers by mimicking human dialogue. They are a crucial part of contemporary applications, enhancing user experiences and facilitating effective collaboration across a variety of fields.

## 2. Literature Review

Mittal, Mamta, et al. presented a structure and functionalities for a chatbot in 2021 [1] using internet technologies. The bot engine was built using a variety of machine learning approaches, such as gradient descent (GD) and natural language processing (NLP) algorithms. The GD algorithm was successively applied to each smaller word batch after the learned data input by the bot was broken into smaller word batches. Less readable by humans are the NLP approaches

 

used to dissect a word into its component parts with a text result.

Ye, Byeong Jin, and others, 2021 [2], Although it is crucial, workers' general health examination (WGHE) follow-up management is currently not done adequately. Although chat-bots, a sort of digital healthcare technology, are employed in many medical specialties, they have never been created in Korea for the follow-up management of WGHE. Kakao Talk and Web Chat were used as user channels in the small-scale implementation of a chat-bot on the Amazon cloud service (AWS) EC2. In terms of effectiveness, 11 (47.83%) gave the usefulness of chatbots a low rating, compared to 21 (91.30%) who gave it a very good rating. 14 (or 60.87%) of the 23 participants said they were satisfied overall.

According to Angappan, A. Kumaresan, et al.'s research from 2021 [3], 60% of patients consult doctors for common health problems, although 80% of such illnesses can be treated at home. People can prevent needless trips to clinics and hospitals by adopting a health chatbot. Thus, a well-designed and implemented chatbot can assist patients who are in remote locations by offering home cures and preventative measures.

According to Sophia, J. Jinu, et al. (2020) [4], conversational bots have developed into a standard user interface for many software services. Conversational bots can be quite helpful when we need a machine to interact with us in a human-like way. These conversational chatbots provide incredibly dependable and affordable service, especially in the management of health. Users frequently aren't aware of all the disease's signs or therapies. For a checkup on minor difficulties, the user must travel to the hospital physically, which takes more time. The processing of complaints received through phone is also extremely hectic. A medical Chatbot can assist in resolving this problem by giving relevant recommendations for healthy living.

Aqib Anwar, Hashim Khan, Jameel, and Umar. (2021) [5], The suggested solution will be created utilizing natural language processing with artificial intelligence, which means the chatbot will initially diagnose the patient's issue before recommending a doctor for consultation. People are interacting with robots and virtual assistants more frequently now than they were before the COVID outbreak.
Robots are used as hospital receptionists in various nations to assist patients and perform human-like tasks. Using chatbots, you can quickly discover a certain doctor by explaining your situation. Making a healthcare chatbot is part of this research effort that includes many literature reviews.

## 3. Research and Methodology

The suggested system presents a sophisticated chat interface with a user-friendly design that enables smooth engagement through the input of symptoms or health-related questions. The system uses machine learning classification methods, particularly the Random Forest Classifier, to predict possible diseases with exceptional precision of 98.43%. The system effectively translates users' spoken input into textual format for processing, even when they choose speech-based communication. A user-centric experience is ensured by this combination of cutting-edge technology, improving accessibility and precision in disease prediction and information distribution.

To determine whether the keywords match a user query or set of user symptoms connected to health, the system performs tokenization, stemming, and keyword extraction on the processed corpus. The system does disease diagnosis and presents the relevant disease description and any precautions or steps the user can take if the user's symptoms rise above the threshold value of four. The system categorizes user input as a health-related question if it is unable to extract any symptoms from the user's input. It then uses the TF-IDF and Cosine Similarity algorithms to select the most pertinent answer from the knowledge database. The completed response is then delivered to the user as text or speech and translated once again into their choice language using the Googletrans python package depending on their preferred mode of communication.

However, other more straightforward systems look for the keyword inside the text and then offer a response based on the matching keywords or established patterns. Most chatbots interpret user input using natural language processing. Today, chatbots can be accessed through numerous firms' applications, websites, and instant messaging systems. They are a part of virtual assistants like Google Assistant. Examples of non-assistive applications include chatbots that are used for research, amusement, and social media campaigns for a particular candidate, product, or subject..

## 4. Proposed Work

### 1. Architectural Diagram

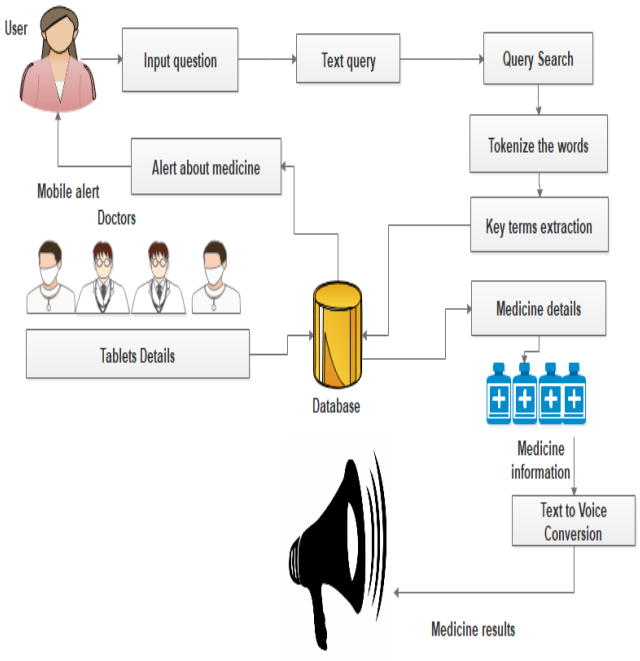


Fig 1: Architectural Diagram

 

The development of systematic methods to define, record, and portray system architectures is a common focus of research efforts in this field. The complex interactions between system components, functionality, and their interdependencies are modeled and analyzed using proven methodologies. Our understanding of how various system components work together to produce desired results is improved by this approach.

Architecture description languages (ADLs) are becoming an important part of technique and research. Researchers and engineers can more easily discuss, analyze, and reason about complex systems because to these formal languages' standardized means of representing system designs. Researchers may exactly simulate, analyze, and optimize systems by using ADLs to precisely represent system components, their attributes, and dynamic behaviors. Overall, system architecture research and methodology concentrate on creating tools, methods, and languages that make it possible to clearly visualize, analyze, and design complex system structures and behaviors. This methodical approach helps create reliable, effective, and well-organized systems across a variety of fields.

**2. Flow diagram**

**LEVEL 0**

The user login and admin response are explained in this DFD diagram.

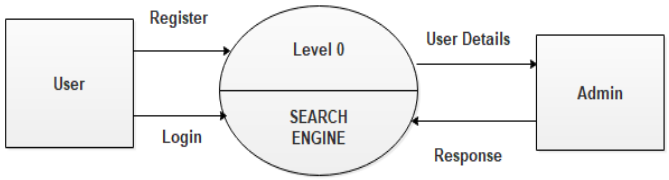


Fig 2: DFD Level 0

**LEVEL 1**

The login details and the question details are stored in the respective database.

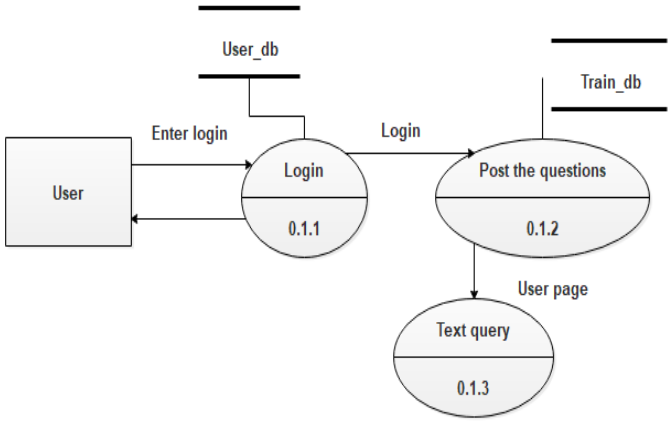


Fig 3: DFD Level 1

**LEVEL 2**

The admin query is tokenized, and the noise words are removed, and preprocessed using keyword, and the response is sent.

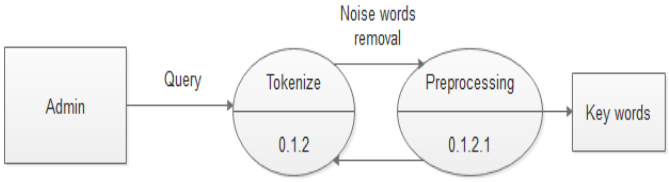


Fig 4: DFD Level 2

**LEVEL 3**

Using the keywords, the question is answered, the answer is converted from text to voice.

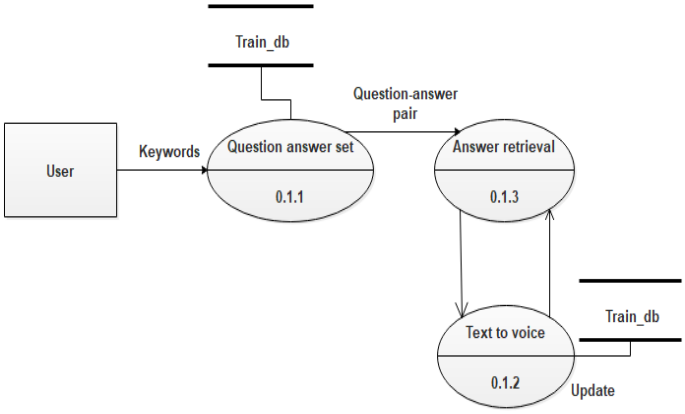


Fig 5: DFD Level 3

## 5. System Implementation

The project's implementation phase entails turning the theoretical design into a workable and efficient system. This phase is crucial for guaranteeing the new system's successful integration and giving end users faith in its effectiveness and efficiency. The process of implementation necessitates careful preparation, considering the limitations of the existing system, and developing plans for a smooth transition. The following lists the crucial actions that make up the implementation process. During the implementation phase, we're putting our plans into action. This is like the moment of truth that shows users our system can really work well for them. By following the steps we've designed, we make sure that technology, ease of use, and practicality all come together smoothly.

Think of it as making sure our system fits the needs of doctors and patients and makes their healthcare experience better. We're taking the steps we've planned and using them to bring our idea to life, making sure everything works just right.

**1. Data Pre-processing**

The initial dataset, which came from the healthcare industry, had 4920 entries with details on 41 different diseases, their symptoms, and safety measures. The goal of the data pre-processing phase was to make sure that the data were consistent and ready for future analysis. The dataset's errors were discovered and corrected. By utilizing the count vectorizer technique, unstructured category data were transformed into structured numerical data. A more refined dataset because of this transformation has each disease

 

represented as a row and each symptom as a column. The binary values (0 or 1) in the dataset represented whether a symptom of a certain disease was present or absent. This data transformation made analysis simpler and allowed machine learning algorithms to be used to forecast diseases from symptoms.

**2. Pre-process data**
The dataset must be cleaned up and normalized during data pre-processing. Information that was redundant or irrelevant was eliminated, and data was standardized to maintain uniformity. The quality of the data used to train and test the machine learning model must be ensured by this phase.

**3. Develop a natural language processing (NLP) model**
The main goal of the project is to create a natural language processing model that can understand user inquiries and provide conversational responses. To help the model comprehend the subtleties of natural language input and produce appropriate replies, NLP approaches were used.

**4. Train the model**
Using the pre-processed data, the machine learning model was trained to look for patterns and relationships between symptoms and diseases. The dataset was split into training and testing sets in order to thoroughly evaluate the model's performance. Numerous machine learning methods, including decision trees, random forests, and neural networks, were considered in order to discover the most effective approach for the task.

**5. Test and refine the model**
The trained model underwent extensive testing to evaluate its recall, accuracy, precision, and overall performance. The model's hyper parameters were adjusted to improve prediction accuracy and boost performance. The goal of this iterative procedure was to develop a solid and trustworthy system for symptom-based disease prediction.

**6. Model Integration and Deployment**
The final stage is integrating the machine learning model into the chatbot system after it has been trained and improved. This integration makes sure that the chatbot can accurately detect diseases based on user-provided symptoms by leveraging the model's predictive skills. Additionally, the system's integration to a web platform or mobile application enables users, including medical professionals and patients, to access the chat bot and take advantage of its features in real-time.

**7. Ethical Considerations and Data Privacy**
Careful consideration of ethical issues and data privacy is required when implementing a system that works with medical data. It is crucial to ensure adherence to medical laws and standards, such as HIPAA. To protect sensitive patient information, the system needs to be constructed with strong security features. To build confidence and adhere to regulatory obligations, it is essential to communicate with users clearly regarding data usage, storage, and confidentiality.

## 6. Project Approach and Functionality Explanation

A chatbot system is implemented in the Flask web application with the source code provided. It provides numerous ways to access the homepage, doctor login, user login, and chat interfaces, among other pages of the program. To achieve its functionality, it makes use of a number of libraries and an external API technique. A web-based chatbot application that enables user-chatbot conversations and can answer questions about health. The application combines the ChatterBot library for chatbot capabilities and uses the Flask framework for web development. For various user tasks, such as checking in as a doctor or user, creating new doctor or user accounts, and having chatbot discussions, the system provides distinct web pages.

**1. Importing Libraries:**
The code starts by importing the necessary libraries and modules that are required and will be utilized by the program.

```
from flask import Flask, render_template, request, session, send_file, jsonify
from werkzeug.utils import secure_filename
import mysql.connector
from chatterbot import ChatBot
from chatterbot.trainers import ListTrainer
from requests import get
from bs4 import BeautifulSoup
```

- Flask: A framework for building and controlling web applications.
- ChatterBot: Offers chatbot features, such as training and user input response.
- Connector for MySQL: Facilitates communication with a MySQL database.
- Requests: Employed to send HTTP requests to outside APIs.
- Web scraping and data extraction from HTML documents are made easier by Beautiful Soup.

**2. Setting Up the ChatBot:**
A ChatterBot instance, an AI chatbot framework, is then created by the code.

```
english_bot = ChatBot('Bot',

storage_adapter='chatterbot.storage.SQLStorageAdapter',
 logic_adapters=[
     {
       'import_path': 'chatterbot.logic.BestMatch'
      },
 ],
 trainer='chatterbot.trainers.ListTrainer')
english_bot.set_trainer(ListTrainer)
```

A bot in English A SQLStorageAdapter is installed on the ChatterBot instance to store chat history. The BestMatch logic adaptor is used to choose the chatbot response. The

 

chatbot can learn from a list of predetermined dialogues by using the ListTrainer as its training mechanism.

**3. Creating the Flask Application:**
The Flask application is created using the following code:

```
app = Flask(__name__)
```

4. **Defining Routes:**
The application specifies a number of routes that link to various chatbot system pages. The URL paths for these routes are specified using the @app.route decorator.

```
@app.route("/")
def homepage():
    return render_template('index.html')

@app.route("/Home")
def Home():
    return render_template('index.html')

@app.route("/DoctorLogin")
def DoctorLogin():
    return render_template('DoctorLogin.html')

@app.route("/NewDoctor")
def NewDoctor():
    return render_template('NewDoctor.html')

@app.route("/chat")
def chat():
    return render_template('chat.html')

@app.route("/UserLogin")
def UserLogin():
    return render_template('UserLogin.html')

@app.route("/NewUser")
def NewUser():
    return render_template('NewUser.html')
```

- The homepage() and Home() functions are used to show the application's main page so that users may start interacting. This is where the portal begins, and as seen in Fig.6, it displays the interface that users see when they first access the portal.
- DoctorLogin() - The login page created for registered doctors is displayed by DoctorLogin. Doctors can access their various profiles and functionality by logging into their accounts through this page, as shown in Fig. 14.
- NewDoctor() - Displays the page for entering the necessary information when creating a new doctor account.
- Chat() - Fig. 11 shows the chat bot UI that is available on this route. Users can interact with the chat bot using text-based dialogues or speech-to-text technology, offering an interactive platform for questions and responses.
- UserLogin() - Fig. 9 Shows the login screen for logged-in users so they can access their accounts.
- The NewUser() - function offers a page where new user accounts can be created by entering the required data shown in Fig. 8.

**5. Running the Application:**
Finally, the application runs using the following code:

```
if __name__ == "__main__":
    app.run(debug=True)
```

Run the script immediately to complete the project. The if __name__ == "__main__": block makes sure that when the script is executed, the Flask development server launches. Based on the stated routes, the server enables users to visit the web pages, communicate with the chatbot, and take numerous other tasks.

**6. Training ChatterBot with ListTrainer**
The provided code sample contains a number of crucial steps for utilizing the ListTrainer method to train a ChatterBot instance. It starts by importing the required modules from the built-in OS module and the ChatterBot library. The next step entails using a try-except block to determine whether the SQLite database file "db.sqlite3" exists.

```
from chatterbot import ChatBot
from chatterbot.trainers import ListTrainer
import os

try:
        os.remove("db.sqlite3")
        print("Old database removed. Training new database")
except:
        print('No database found. Creating new database.')
english_bot = ChatBot('Bot')
english_bot.set_trainer(ListTrainer)
for file in os.listdir('data'):
    print('Training using '+file)
    convData = open('data/' + file).readlines()
    english_bot.train(convData)

    print("Training completed for "+file)
```

This method makes sure that all previously trained data is removed, leaving a blank slate for future training. Then, a fresh ChatBot instance with the name "Bot" is created. The ListTrainer training strategy, which enables training based on lists of pre-defined talks, is used in the code as the chosen training approach. Using the os.listdir('data') function, the process' central loop iterates through each file in the chosen "data" directory. The code opens each file throughout this loop and reads each line of data inside. The variable convData is then used to hold each line that represents a discussion. After that, the 'english_bot' ChatBot instance is trained using the chat data from the file. The code emits a message after the training process for each file is finished, indicating that the training was successful for that particular file.

 

### 7. Using YAML Configuration

To train the chatbot, I used a complicated strategy. For training, I use a YAML file rather than manual intervention. The human-readable data serialization format YAML (YAML Ain't Markup Language) is frequently used for configuration files. I clearly and methodically organize the interactions and responses I gain from using a YAML file.

My system's independent collecting of patient information is one of its distinctive features. This indicates that I am able to get pertinent information from the patients without a doctor's intervention. By streamlining the data collection procedure, this method benefits both patients and medical staff by increasing efficiency.

In my methods, I make use of tokenization, stopword elimination, and stemming algorithms.

- **Tokenization** is a crucial step in which I divide a given text into manageable chunks called tokens. These "tokens," which may be single words or even phrases, would help me better evaluate and comprehend the text's constituent parts. I can better understand the input data's structure using this technique.

- **Remove Stopwords**: Stopwords are often used words in a language that have little or no significance, such as "the," "and," "is," and so on. I filter these words from the incoming data using stopword removal. By doing this, I concentrate on the words that have a greater contextual significance, which helps me understand the user's query. process of reducing words to their fundamental or basic form. For instance, the fundamental form of terms like "running," "runs," and "ran" is "r

- **Stemming** is the un." This method enables me to treat word variations as a single entity, improving the precision of text analysis.

### 8. Generating Precise Responses:

My machine learning component kicks in after the user's input has been examined and matched with pertinent patterns and context. I'm able to produce responses that are relevant to the context and in line with the user's query based on the matching procedure.

By combining contextual word matching, machine learning algorithms, and the structured training from the YAML file, I am able to respond to patient questions with accuracy and relevance without the need for direct physician intervention. This method streamlines the process of gathering medical data while simultaneously enhancing the user experience.

## 7. Results

### 1. Main Page

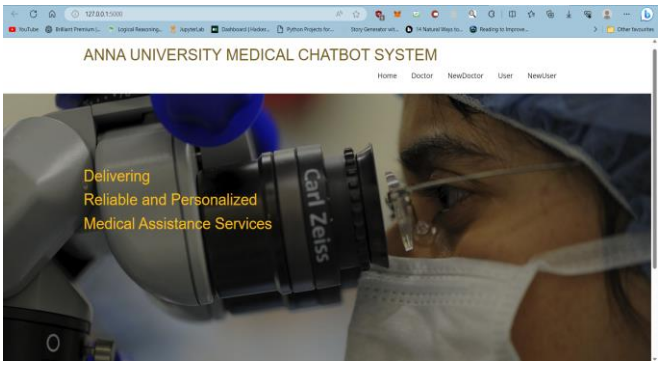


Fig 6: Main Page

### 2. Doctor Registration

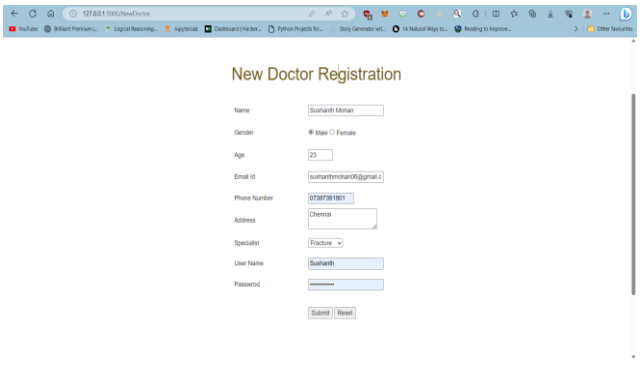


Fig 7: New Doctor Registration

### 3. User Registration

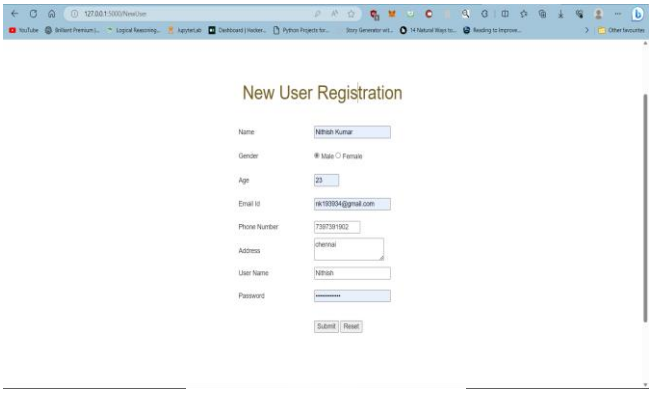


Fig 8: New User Registration

### 4. User Login

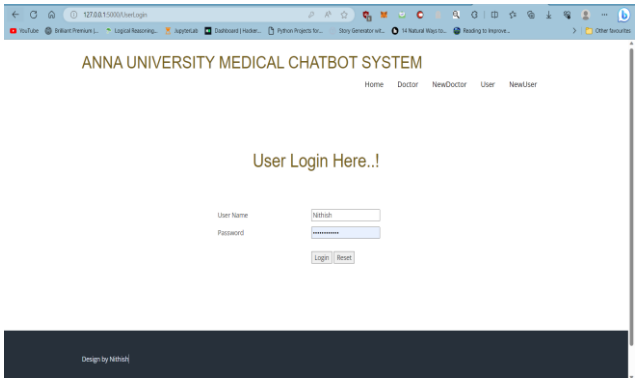


Fig 9: User Login

 

### 5. User Personal Information

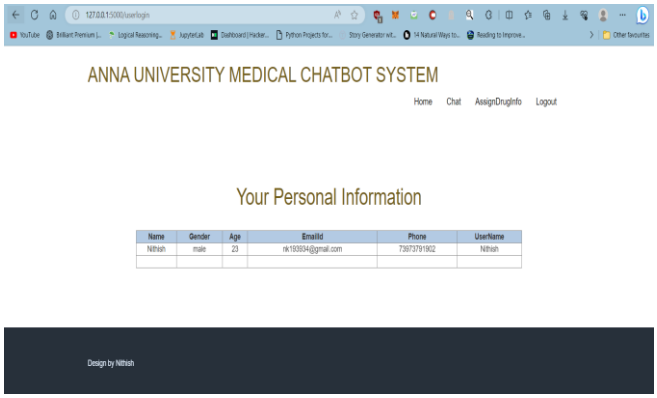


Fig 10: User Personal Information

### 6. Healthcare Chatbot

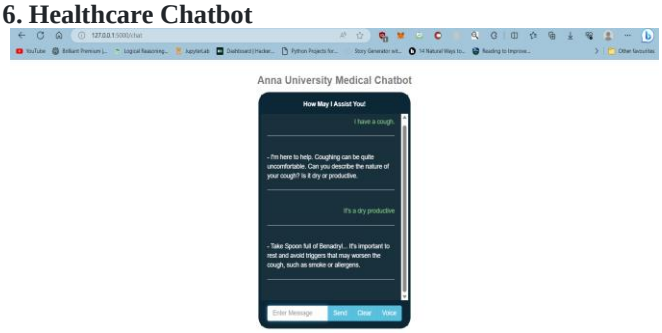


Fig 11: Healthcare Chatbot

### 7. Healthcare Chatbot

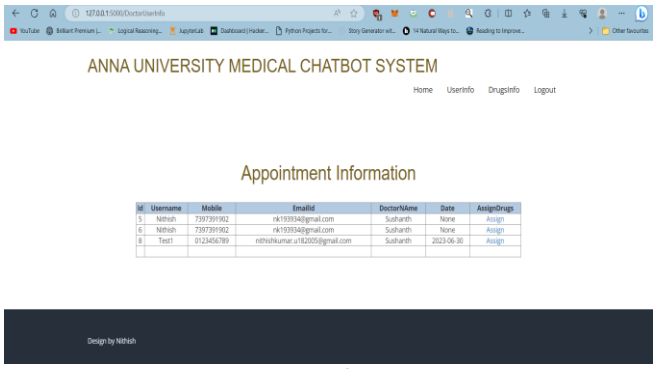


Fig 12: Doctor Appointment

### 8. Doctor Appointment Record

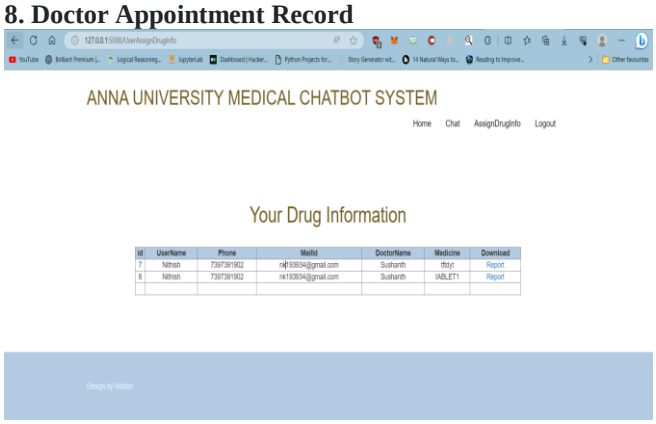


Fig 13: User Drug Information

### 9. Doctor Login

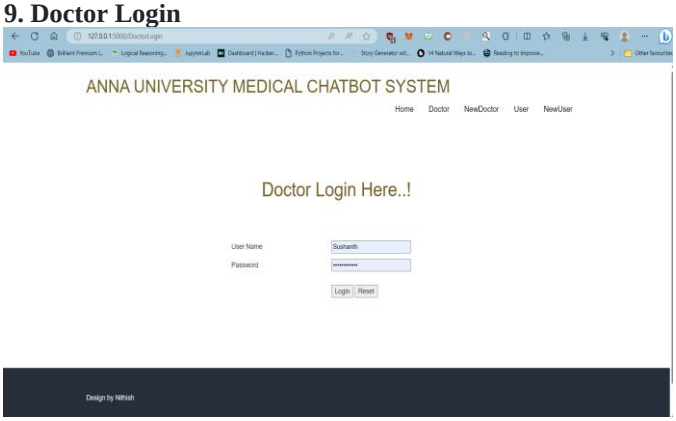


Fig 14: Doctor Login

### 10. Doctor Personal Information

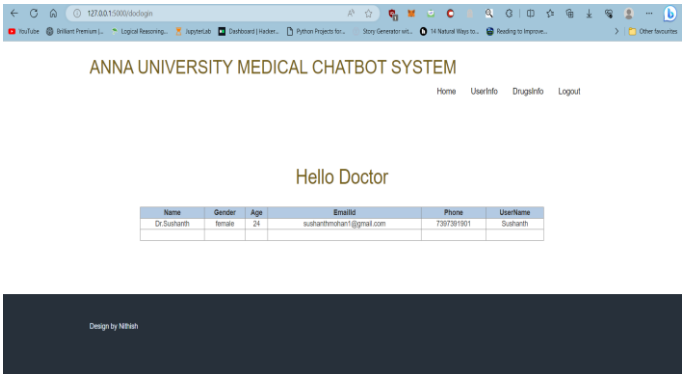


Fig 15: Doctor Personal Information

### 11. Appointment Information

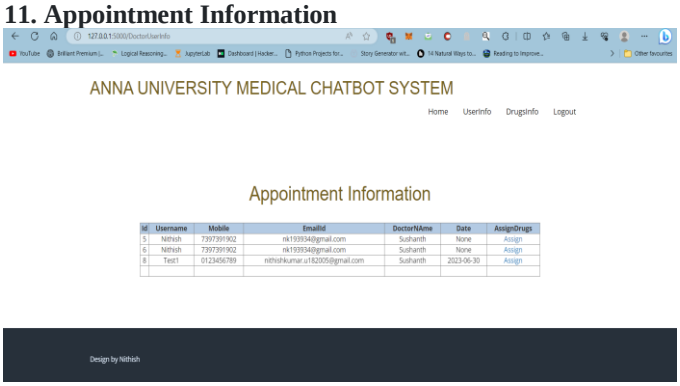


Fig 16: Appointment Information

### 12. Assign Drugs to Patient

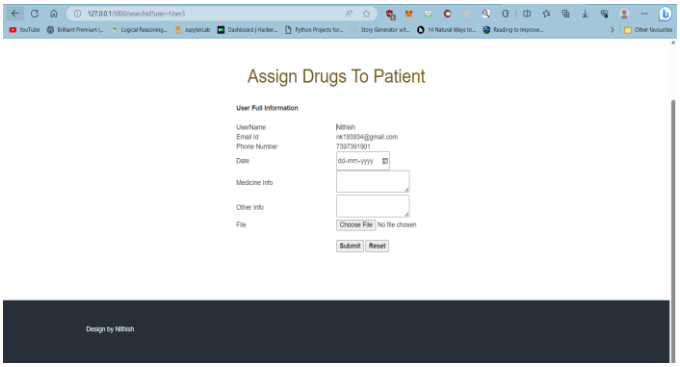


Fig 17: Assign Drugs to Patient

 

## 8. Conclusion

In conclusion, our technology offers consumers the comfort of asking questions about medical dose via voice commands, introducing a significant resource for medical institutions and hospitals. The system effortlessly retrieves drug-related data from the medication API, vocalizes the findings, and displays them for quick access. Natural Language Processing (NLP) integration allows the system to communicate with users in a language they are acquainted with, facilitating effective interaction and improving user experience.

The surge of complex and varied data brought on by healthcare transactions highlights the value of using cutting-edge methods like data mining. Our method successfully uses data mining to glean insightful information from the massive volume of medical data, potentially yielding ground-breaking findings that could save lives. Medical chatbot adoption represents a crucial step towards individualized healthcare. These chatbots protect patient privacy while convincingly imitating intelligent interaction by utilizing exclusive AI/ML technology and enabling natural, intent-driven dialogues. Users can experience the advantages of AI-driven conversational technology with as little investment as a simple doctor appointment chatbot.

Essentially, our system is a demonstration of how data mining, NLP, and AI have the potential to fundamentally alter how healthcare professionals diagnose patients and communicate with one another. As technology develops, the interaction of intelligent systems and humans has the potential to change the healthcare sector, enhancing outcomes and experiences for all parties.

## Reference


[1] Mittal, Mamta, Gopi Battineni, Dharmendra Singh, Thakursingh Nagarwal, and Prabhakar Yadav. "Web-based chatbot for frequently asked queries (FAQ) in hospitals." *Journal of Taibah University Medical Sciences* , Vol.**16**, Issue.**5**. pp.**740-746, 2021.**

[2] R. Singh, M. Paste, N. Shinde, H. Patel, and N. Mishra, "Chatbot using TensorFlow for small Businesses," *in Proc 2018 Second International Conference on Inventive Communication and Computational Technologies (ICICCT)*, Coimbatore, India, pp.**1614-1619**, **2018**. Doi: 10.1109/ICICCT.2018.8472998.

[3] R. B. Mathew, S. Varghese, S. E. Joy and S. S. Alex, "Chatbot for Disease Prediction and Treatment Recommendation using Machine Learning," *in Proc 2019 3rd International Conference on Trends in Electronics and Informatics* (ICOEI), Tirunelveli, India, pp.**851-856**, **2019.** Doi: 10.1109/ICOEI.2019.8862707.

[4] M. C. Surabhi, "Natural language processing future," 2013 International Conference on Optical Imaging Sensor and Security (ICOSS), Coimbatore, India, pp.**1-3, 2013**. Doi: 10.1109/ICOISS.2013.6678407.

[5] F. M. Barcala, J. Vilares, M. A. Alonso, J. Grana and M. Vilares, "Tokenization and proper noun recognition for information retrieval," Proceedings. 13th International Workshop on Database and Expert Systems Applications, Aix-en-Provence, France, pp.**246-250, 2002.** Doi: 10.1109/DEXA.2002.1045906.

[6] D. Madhu, C. J. N. Jain, E. Sebastain, S. Shaji and A. Ajayakumar, "A novel approach for medical assistance using trained chatbot," 2017 International Conference on Inventive Communication and Computational Technologies (ICICCT), Coimbatore, India, pp.**243-246, 2017.** Doi: 10.1109/ICICCT.2017.7975195.

[7] B. R. Ranoliya, N. Raghuwanshi and S. Singh, "Chatbot for university related FAQs," 2017 International Conference on Advances in Computing, Communications and Informatics (ICACCI), Udupi, India, pp.**1525-1530, 2017.** Doi: 10.1109/ICACCI.2017.8126057.

[8] M. Omar, S. Choi, D. Nyang and D. Mohaisen, "Robust Natural Language Processing: Recent Advances, Challenges, and Future Directions," in IEEE Access, Vol.**10**, pp.**86038-86056**, **2022.** Doi: 10.1109/ACCESS.2022.3197769.

[9] A. Goel, Satyam and S. Sharma, "Artificial Intelligence based Healthcare Chat Bot System," 2023 8th International Conference on Communication and Electronics Systems (ICCES), Coimbatore, India, pp.**1362-1366, 2023.** Doi: 10.1109/ICCES57224.2023.10192727.

[10] T. A. H. Shaikh and M. Mhetre, "Autonomous AI Chat Bot Therapy for Patient with Insomnia," 2022 IEEE 7th International conference for Convergence in Technology (I2CT), Mumbai, India, pp.**1-5, 2022.** Doi: 10.1109/I2CT54291.2022.9825008.

[11] A. K. Singh, "Conversational Interfaces for Health Consultation," 2022 11th International Conference on System Modeling & Advancement in Research Trends (SMART), Moradabad, India, pp.**35-38, 2022.** Doi: 10.1109/SMART55829.2022.10046984.

[12] A. Wahal, M. Aggarwal and T. Poongodi, "IoT based Chatbots using NLP and SVM Algorithms," 2022 3rd International Conference on Intelligent Engineering and Management (ICIEM), London, United Kingdom, pp.**484-489, 2022.** Doi: 10.1109/ICIEM54221.2022.9853095.

[13] R. Goel, R. P. Goswami, S. Totlani, P. Arora, R. Bansal and D. Vij, "Machine Learning Based Healthcare Chatbot," 2022 2nd International Conference on Advance Computing and Innovative Technologies in Engineering (ICACITE), Greater Noida, India, pp.**188-192, 2022.** Doi: 10.1109/ICACITE53722.2022.9823901.

[14] Vanshika and N. Gupta, "Machine Learning Applications in Healthcare," 2022 10th International Conference on Reliability, Infocom Technologies and Optimization (Trends and Future Directions) (ICRITO), Noida, India, pp.**1-6, 2022.** Doi: 10.1109/ICRITO56286.2022.9964865.


**AUTHORS PROFILE**

**Nithish Kumar S** is a BCA graduate currently studying an MCA at ANNA UNIVERSITY BIT CAMPUS, TRICHY, he is a tech enthusiast. Nithish, who exhibits a strong interest in technology, is an expert programmer who is familiar with Python and its API features. His competence in the fields of machine learning and artificial intelligence (AI) reflects his lifelong fascination in cutting-edge technologies. Nithish completed a 4-month internship where he worked heavily with C#, Python, and ASP.Net, and as a result, he has developed his abilities through real-world experiences. His proficiency has been improved through his exposure to a variety of programming languages and frameworks. Nithish, while being young, has a great knack for comprehending and applying ML algorithms, which he uses to advance the field of AI-driven solutions.

 

S. Sujatha is a distinguished academician with a noteworthy career path in the fields of engineering and computer science. She graduated from IIT Bombay with a B. Tech., M. Tech., and Ph.D. in physical science in 1998, 2002, and 2008, respectively. She has had a varied and illustrious career and has influenced both academia and research. Since December 2009, S. Sujatha has held the position of Associate Professor at the University College of Engineering, Tiruchirappalli. She offers a variety of experience to her work. She held lecturing positions at RMK Engineering College in Chennai from September 2004 to August 2005 and at Anna University of Technology, Tiruchirappalli, from June 2007 to December 2009. She also served as a lecturer at Nagamalai Navarasam College of Arts & Science in Erode from May 1995 to April 1996 and at KSR College of Arts & Science in Tiruchengodu, Erode, from October 1997 to September 2004. She has participated in several initiatives and made important contributions to her profession out of a deep commitment to research. Her work has been honored with numerous medals and accolades, and her study focuses on data mining. She has over 15 years of teaching experience and ten years of research expertise. Her extensive amount of work and dedication to the academic community demonstrate her commitment to facilitating learning, developing research, and having a beneficial impact.

 