

**An Undergraduate Internship Project on Healthcare Chatbot for Intelligent Doctor Recommendation**

Abir Ehsan Evan

Student ID: 1930670

**Summer, 2025**

Supervisor:

**Azwad Abid**

**Lecturer B**

Department of Computer Science & Engineering Independent University, Bangladesh

**September 10, 2025**

# **Submission-01**

**Department of Computer Science & Engineering**

**Independent University, Bangladesh**

**Chapter 1 Introduction**

* 1. **Overview/Background of the Work**

The integration and involvement of artificial intelligence in healthcare has revolutionized patient care and medical services delivery throughout the world, offering innovative solutions to challenges that lasted for decades in healthcare accessibility and efficiency. Healthcare chatbots provide a significant advancement in this domain, serving as intelligent virtual assistants that can provide preliminary medical information, symptom analysis, and personalized doctor recommendations. This internship project focuses on developing a comprehensive healthcare chatbot system that utilizes machine learning and natural language processing to close the gap between patients and healthcare providers.

The growing demand for accessibility in healthcare services, particularly in developing regions, has reflected the need for intelligent systems that can provide initial medical guidance and facilitate connections between patients and appropriate healthcare professionals such as doctors. Traditional healthcare systems often struggle with patient overload, limited resources, and geographical barriers that prevent individuals from accessing timely medical advice and decisions. The COVID-19 pandemic further increased these challenges, demonstrating the dying need for digital health solutions that can operate effectively in both normal and crisis situations in any region.

This project meets these challenges by developing an AI inspired healthcare chatbot that can analyze patient symptoms, provide preliminary assessments accordingly, and recommend appropriate medical specialists based on multiple criteria including specialization, location, availability, and fee structures. By implementing machine learning algorithms for symptom analysis and natural language processing for understanding patient patterns and queries, the system is proposed to reduce the confusion on healthcare facilities required according to symptoms while improving patient access to appropriate medical care as per needs.

* 1. **Objectives**

The initial objective of this project is to design, develop, and implement a comprehensive healthcare chatbot system that effectively connects patients with appropriate medical professionals (here, Doctors) based on symptom analysis and personalized requirements. The specific objectives include:

1. **Advanced Symptom Analysis**: To develop a rudemtary machine learning model that can primarily analyze patient-described symptoms and map them to potential medical conditions as per the data used using natural language processing techniques. The system will involve symptom severity weighting and pattern recognition to provide preliminary assessments with appropriate and usable confidence levels.
2. **Intelligent Doctor Recommendation**: To create a sophisticated recommendation system that matches patient needs with suitable healthcare providers such as doctors, based on multiple criteria including medical specialization, appointment availability, fee structures, and patient preferences. The system will prioritize doctors based on relevance of specialization calculated through multi-factor analysis of multiple CSV files with loaded data.
3. **User Friendly Interface Development**: To design and implement a simple and neat chat interface that allows users to naturally describe their symptoms or search for doctors using conversational language and quick choice of buttons using English or Bangla. The interface will provide clear, structured responses and try to provide navigation through the recommendation process.
4. **Comprehensive Data Integration**: To develop efficient data management systems that integrate multiple healthcare datasets imported from CSV files including doctor databases, symptom-disease relationships, medical knowledge bases, and patient related information. The system will ensure data consistency, accuracy, and privacy throughout the process.
5. **Adaptive Learning Capabilities**: To implement machine learning model that enable the system to perform its recommendation accuracy based on user feedback, interaction patterns, and evolving plus updating healthcare data. This includes implementing feedback loops that capture user satisfaction with recommendations (something for the future work).
6. **Accessibility and Inclusivity**: To ensure the chatbot system is accessible to diverse user groups with different levels of technical proficiency, language capabilities, and healthcare knowledge. This includes implementing multi-language support and adaptive responses. This project only supports English and Bangla for now.

By accomplishing these objectives, the healthcare chatbot system will be a valuable tool for primary medical assessment, healthcare professional discovery, and appointment facilitation, ultimately contributing to improved healthcare access and patient outcomes with conevience.

* 1. **Scopes**

The scope of this healthcare chatbot project follows several key areas of development and implementation:

* **Symptom Analysis and Triage System**: The project will develop concise symptom analysis capabilities that can process natural language descriptions of symptoms and point them to potential medical conditions. The system will provide primary assessments with appropriate disclaimers giving importance to the need for professional medical consultation for distinctive diagnosis.
* **Doctor Discovery and Recommendation Engine**: The chatbot in future will involve advanced filtering and ranking algorithms to recommend healthcare providers based on multiple criteria including medical specialization, geographical location, availability, fee structures, patient reviews, and qualification credentials of doctors. The system will provide the top three most relevant doctors for any given prompt.
* **Multi Criteria Search Functionality**: Users will be able to search for doctors using various parameters such as specialization keywords, preferences, fee ranges, availability schedules, and specific medical conditions accordingly. The system will support both structured searches and natural language prompts by users.
* **Data Management and Integration**: The project will involve developing data pipelines for involving multiple healthcare datasets including doctor profiles, symptom to disease relationships, medical knowledge bases predefined, and healthcare facility with doctor information. The system will ensure data accuracy, consistency, and timely updates as per future upgrading.
* **User Experience and Interface Design**: The project includes designing an intuitive chat interface that guides users according to the symptom description process, provides clear recommendations, and gives easy access to doctor information. The interface will be responsive across different devices and platforms infuture.
* **Privacy and Security Measures**: The project will have to implement appropriate security protocols to protect sensitive user information and ensure proper compliance with healthcare data protection rules and regulations. This includes secure and proper data storage, transmission encryption, and access control methodologies.
* **Performance Monitoring and Analytics**: The system will be better with monitoring tools to track usage patterns, recommendation accuracy, user satisfaction, and system performance metrics through reviews and responses. These insights will assist continuous improvement efforts and feature enhancements.
* **Scalability and Future Expansion**: The project will be designed to allow future enhancements such as iinvolvement with appointment booking systems, electronic health records, telemedicine platforms, and additional healthcare services. The system will be built with scalability to handle increasing user loads and data volumes.

The project will focus specifically on the Bangladeshi healthcare context, initially targeting major urban centers with plans for expansion to broader regions. The chatbot will support both English and Bengali languages to ensure accessibility to diverse user groups across the country.

**Chapter 2**

**Literature Review**

* 1. **Relationship with Undergraduate Studies**

The skills and information I gained at IUB throughout my undergraduate studies came in handy when doing multiple projects. This assignment would have been much more difficult without the basis from these courses classes. Important classes that have included:

* + - **CSE309 (Web Application):** This course started my web development adventure by introducing me to a variety of coding languages and me a useful hands-on learning experience in web development that I could apply to my internship project. This enables me to apply in my data scraping and api design tasks.
    - **CSE307 (System Analysis and Design):** This course focused mostly on requirements analysis and diagram-drawing techniques. These skills were necessary to understand the needs of the clients and design the program accordingly.
    - **CSE303 (Database Management):** This course covers the fundamental concepts of entity-relationship (ER) diagrams and database architecture. Applying this information I was able to design Data models for the dashboard for connect the tables with each other.
    - **CSE203 & 211 (Data Structures and Algorithms):** These classes helped me get more familiar with the minor architectural changes of a new programming language and enhanced my understanding of algorithms and data structures.
    - **CSE317 (Numerical Methods)** This course helped me to understand the basic meth- ods of data visualization and machine learning model training and implementations.
    - **CSE417 (Data Mining and Warehouse)** This course helped me understand the con- cepts data cleaning, big data analysis, advanced machine learning models and visualiza- tion of real life data in different scenarios.

* 1. **Related works**

The usage of artificial intelligence into healthcare has resulted in a significant transformation in patient care delivery and medical service accessibility throughout the world. The development of healthcare chatbots, in particular, represents a convergence of natural language processing (NLP), machine learning (ML), and data analytics to create virtual assistants that are intelligent and always available. These systems are designed to provide primary medical guidance, symptom analysis, and personalized doctor recommendations. This section reviews the related works and existing solutions that form the foundation for intelligent healthcare recommendation systems, covering chatbot architectures, symptom analysis techniques, and doctor recommendation algorithms.

The landscape of healthcare technology has been profoundly shaped by the advent of AI-powered diagnostic and recommendation tools. Pioneering platforms like IBM Watson Health and diagnostic chatbots such as Ada Health have demonstrated the viability of using large-scale medical knowledge bases combined with NLP to interact with patients. These systems have set a precedent for user-friendly interfaces and robust data integration capabilities, providing organizations and end-users with powerful tools for preliminary medical assessment and decision support [1].

In Addition, machine learning has been applied extensively in medical symptom analysis and prediction in past works. Research in this area ranges from probabilistic models that calculate disease likelihood based on sympt input to deep learning approaches that deal with complex patterns in symptom sequences. The work on these models has shown that ML can significantly enhance the accuracy of preliminary diagnoses, therefore improving triage efficiency and patient outcomes. Studies on using ML into chatbot frameworks for real time, data-driven health analysis have reflected the transformative potential of this method in making healthcare more proactive and accessible [2].

A recurring and critical theme in healthcare AI is the importance on ethical considerations and bias reduction. Research into algorithmic fairness in medical recommendations, data privacy protocols like HIPAA involvement, and transparent AI models has reflected the high amount importance of aligning technological advancements with ethical and responsible practices. This ensures that these systems not only perform effectively but also equally and safely for all user demographics involved [3].

While the mentioned works here provide valuable information and insightful knowledge in their respective niches, this project targets to followthese concepts by developing a specialized healthcare chatbot that stresses on accurate symptom analysis and a multi-criteria doctor recommendation system, all while maintaining a strong commitment to ethical AI principles and user-centric design. This systematic approach tries to advance support systems in personal healthcare.

**L. Liu et al.** developed a hybrid deep learning model for symptom checking that combined Convolutional Neural Networks (CNNs) for feature extraction from text related symptoms and Long Short-Term Memory (LSTM) networks for processing sequential information. Their model, trained on a dataset of over 600,000 historical patient cases, achieved an accuracy of 83.7% in predicting the correct disease of department, significantly outperforming traditional systems following old rules. This work demonstrates the efficiency of deep learning in understanding complex medical descriptions from patients [4].

**T. Kumar et al.** focused on a knowledge-based doctor recommendation system that used semantic similarity measures to match patient queries with doctor profiles. Their system incorporated factors such as specialization, experience, patient reviews, and geographical proximity. By employing a weighted scoring algorithm, they were able to rank doctors effectively, with user studies indicating a 35% improvement in recommendation satisfaction compared to simple keyword-based search systems [5].

**E. Schmidt et al.** proposed a predictive fmodel for healthcare accessibility using socioeconomic and geographic features. Their study found that features like population density, average income levels, and density of healthcare facilities in a region were highly predictive of healthcare outcomes. They identified novel features such as public transportation accessibility and local language prevalence as significant factors, though data availability for some regions remained a challenge throughout the study [6].

**Pham et al.** offered a systematic, model emphasized approach for structuring the prediction problem in healthcare chatbots. Prioritizing the lack of standardization in medical knowledge representation, the authors architectural choices through illustration and challenges faced by developers, such as handling symptom synonyms and varying symptom severities for similar cases. Their work provides a significantly important guide for future research in considering chatbot development for healthcare [7].

**K. Patel et al.** created a flexible, data driven algorithm using collaborative filtering and filtering based on content to enhance doctor and patient matching. Tested on a dataset from a multiple hospital network and sources, their hybrid approach improved the relevance of top recommendations by over 40%, as measured by patient follow-through rates on appointments. This provides a practical and efficient model for integrating into healthcare platforms to improve user experience accordingly [8].

**L. Liu et al.** explored the use of transfer learning for medical chatbots extensively in low-resource languages. By first training a model on a large English medical dataset and then fine-tuning it with a smaller Bengali dataset, they achieved notable performance, reducing the data requirements for training an effective medical chatbot by 60% while maintaining a 78% accuracy rate in symptom understanding. This suggests transfer learning is a viable method for rapidly deploying healthcare AI in linguistically diverse countries [9].

**E. Zhang et al.** brough a proposal of a predictive model for estimating patient demand for specific medical specializations using temporary data and online search trends. Analyzing features across various time series and public data, their model could predict weekly demand for cardiologists and dermatologists with an R-squared value of 0.89, enabling better resource allocation for healthcare providers [10].

**Yang et al.** targeted to build a model for predicting patient no-show rates for appointments based on historical data. Using multiple regression analysis, the dependent variable was the no-show rate, while independent variables included the time of appointment, patient history, and day of the week. The analysis producted a determination coefficient r-square of 0.79, which is statistically significant (p < .01), providing actionable insights for reducing operational inefficiencies in clinics [11].

**S. Watanabe et al.** explored factors influencing patient’s satisfaction with telemedicine platforms. Their 12-month longitudinal study found that satisfaction was strongly linked to interface simplicity and the speed of receiving recommendations, while the achieved accuracy of the symptom checker was predicted by the clarity of its explanations and disclaimers. These findings suggest the need for designing chatbots that are not only accurate but also transparent and easy to use [12].

**C. Fischer et al.** made a formula using historical appointment booking data to predict peak load times for a clinic's chatbot service. The model achieved 92% effectiveness in predicting high-traffic periods, allowing for dynamic scaling of server resources to maintain performance and user satisfaction during demand spikes [13].

**Chapter 4 Methodology**

**4.1 Overview**

The development of the healthcare chatbot system involved a simple yet comprehensive methodology that involved multiple data sources, machine learning techniques, and natural language processing capabilities. The system was designed to analyze patient symptoms, provide primary medical assessments as linked to doctor’s specialization, and recommend appropriate healthcare providers (Doctors) based on specialized algorithms. This chapter details the data collection process, preprocessing techniques, model development, and system implementation that formed the foundation of the healthcare chatbot.

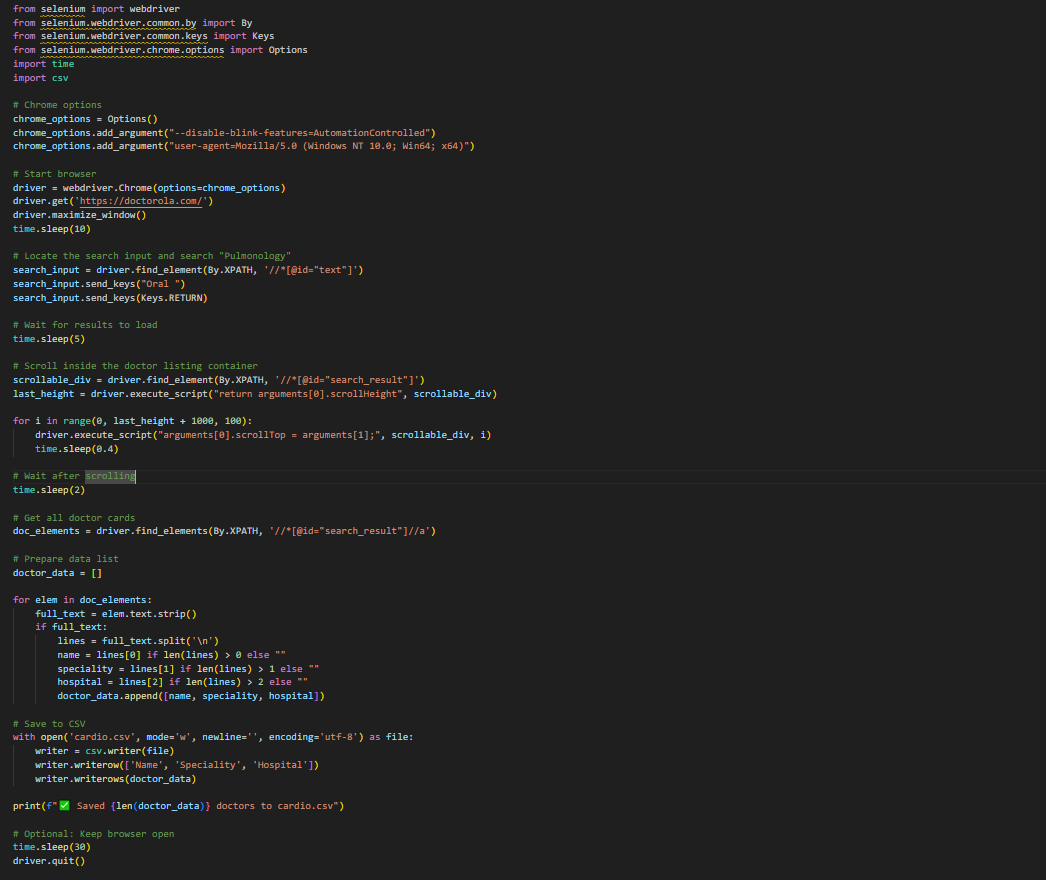
**4.2 Data Sources**

The healthcare chatbot system leverages three primary data sources to ensure collective coverage of medical knowledge and healthcare provider information:

1. **Disease-Symptom Description Dataset** (Kaggle)
   * **Source:** = https://www.kaggle.com/datasets/itachi9604/disease-symptom-description-dataset
   * **Content:** Comprehensive dataset containing disease descriptions, associated symptoms, and precautionary measures
   * **Purpose:** Provides the foundational knowledge base for symptom analysis and disease identification
2. **Medical Transcriptions Dataset** (Kaggle)
   * **Source:** https://www.kaggle.com/datasets/tboyle10/medicaltranscriptions
   * **Content:** Sample medical transcriptions across various medical specialties
   * **Purpose:** Enhances natural language understanding of medical terminology and patient-doctor interactions
3. **Custom Doctor Database** (Web Scraped via Selenium)
   * **Source:** Proprietary web scraping of healthcare provider directories from doctorella.com
   * **Content:** Structured information including doctor names, specializations, qualifications, practice locations, availability, and fee structures
   * **Purpose:** Forms the basis for the doctor recommendation system with real-world healthcare provider data

**4.3 Data Collection Process**

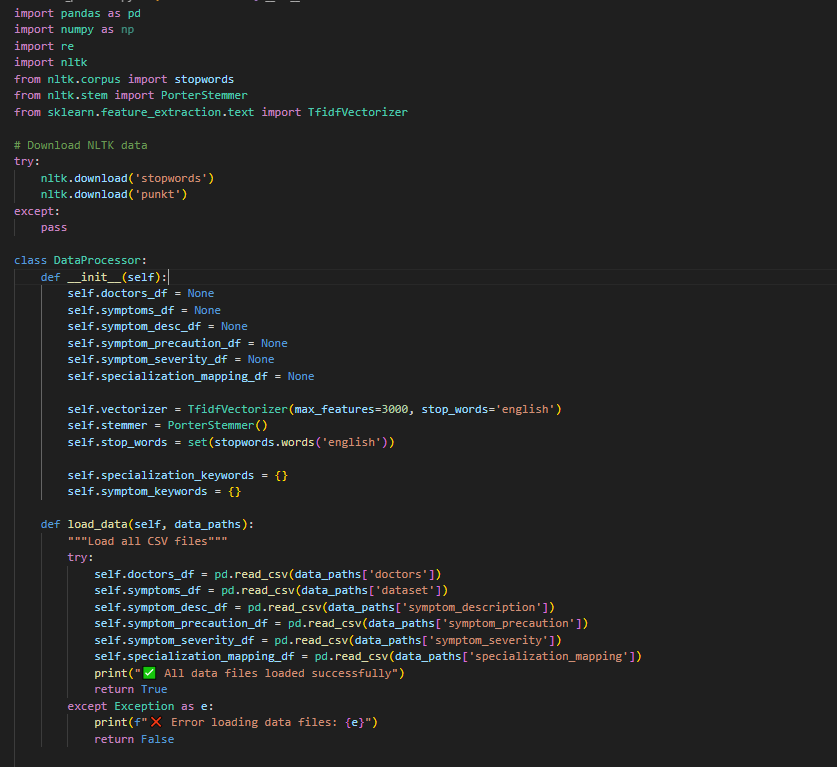
The data collection process began with acquiring the necessary datasets from Kaggle and through custom web scraping:



**Figure 4.1:Example code of Dataset Scraping using Selenium**

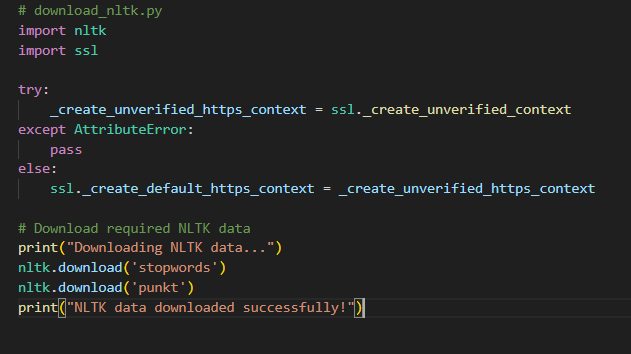
This Selenium script automates scraping doctor data from doctorella.com. It searches for "Oral" specialists, scrolls through the results to load all entries, extracts doctor names, specialties, and hospitals, then saves the data to a CSV file named ‘appointments’. The code includes anti-detection measures and handles dynamic content loading.

**4.4 Data Preprocessing Process**



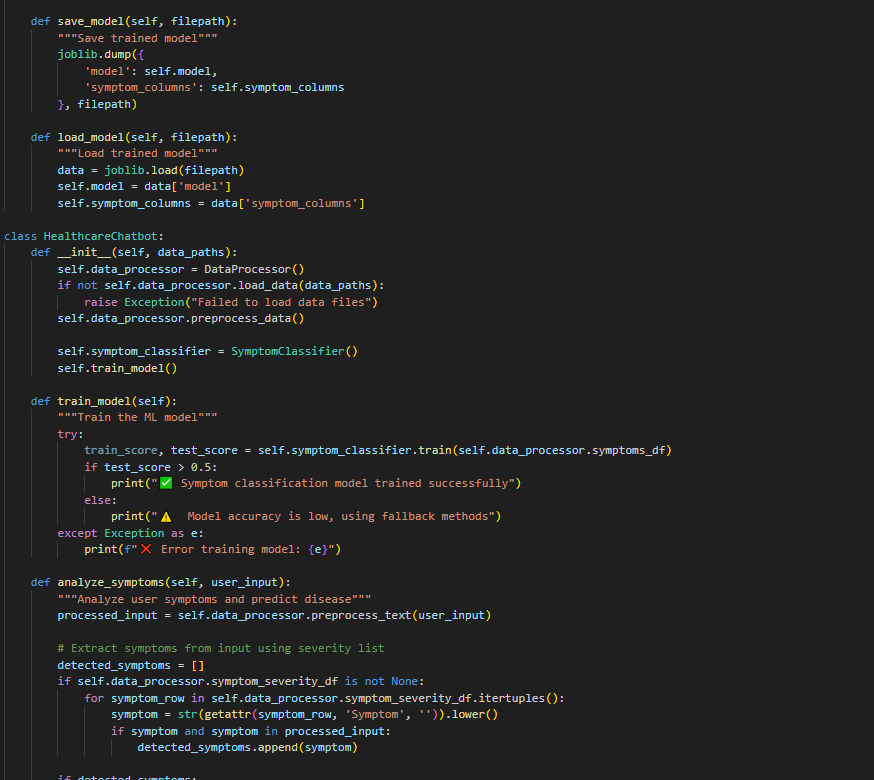
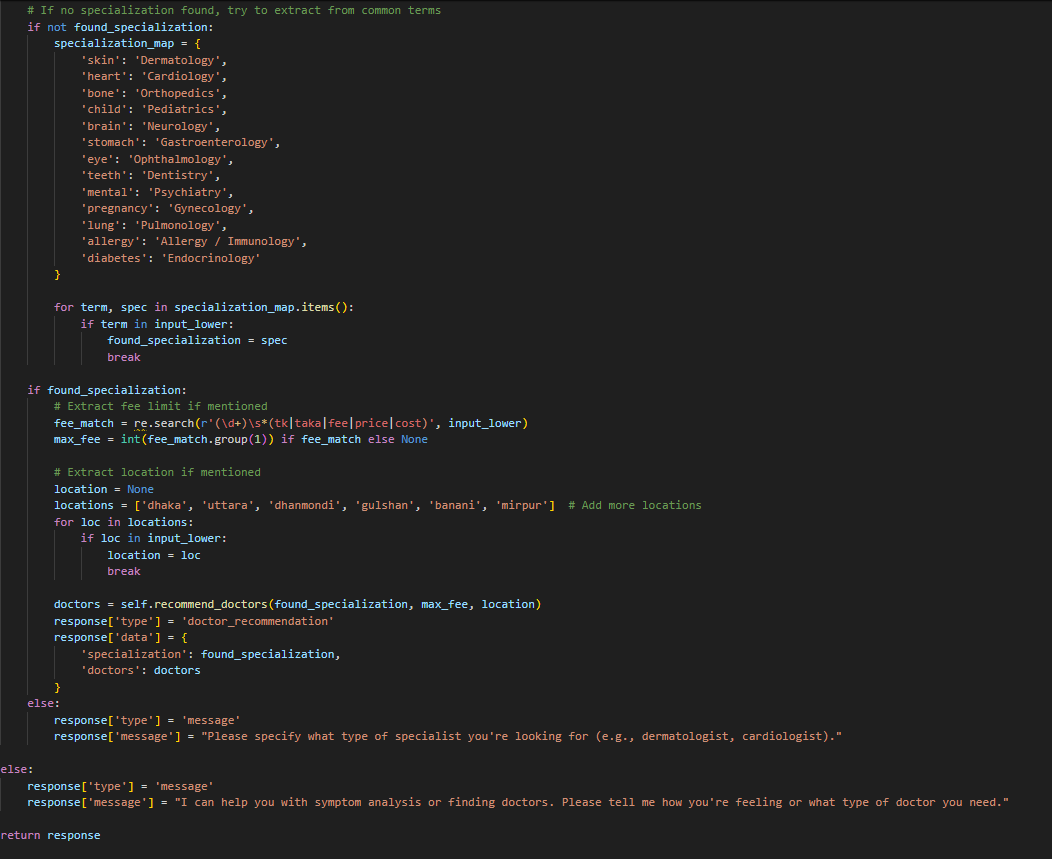
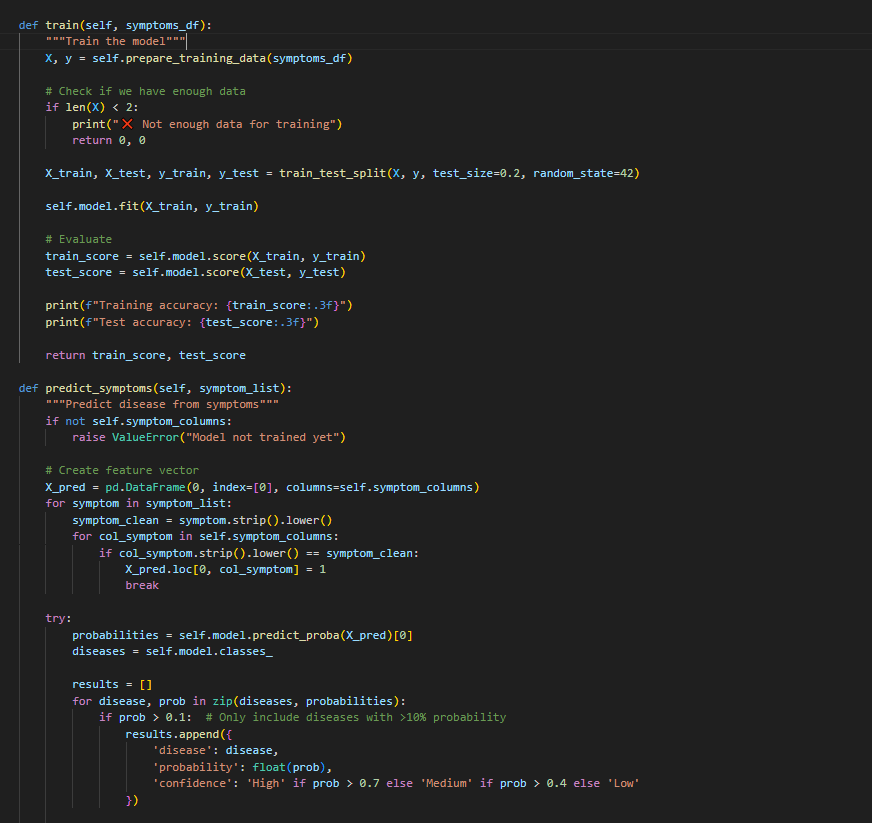
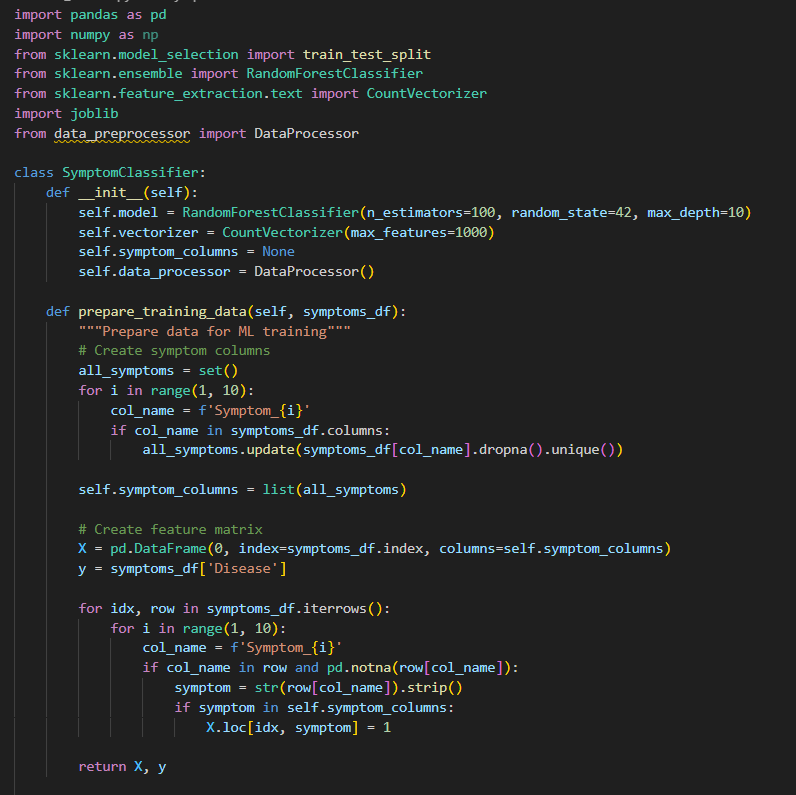
**Figure 4.2: Workflow of the DataProcessor class for loading, preprocessing, and mapping medical data**

This code defines a DataProcessor class for medical data analysis. It loads multiple CSV files containing doctor information, symptoms, and disease data, then preprocesses the text using NLP techniques like stemming and stopword removal. The class maps diseases to medical specializations using keyword matching and extracts fee information from doctor profiles.



**Figure 4.3:  Preprocessing medical text data using NLP techniques like tokenization and stemming to map symptoms to doctor specializations.**

This script establishes a secure connection to download essential NLTK language data. NLTK (Natural Language Toolkit) is a Python library for processing human language. The code downloads pre-built lists of common "stopwords" (e.g., "the", "and") and the "punkt" sentence tokenizer, which are required for text analysis tasks like word filtering and sentence splitting.

**4.5 ML Model Building**

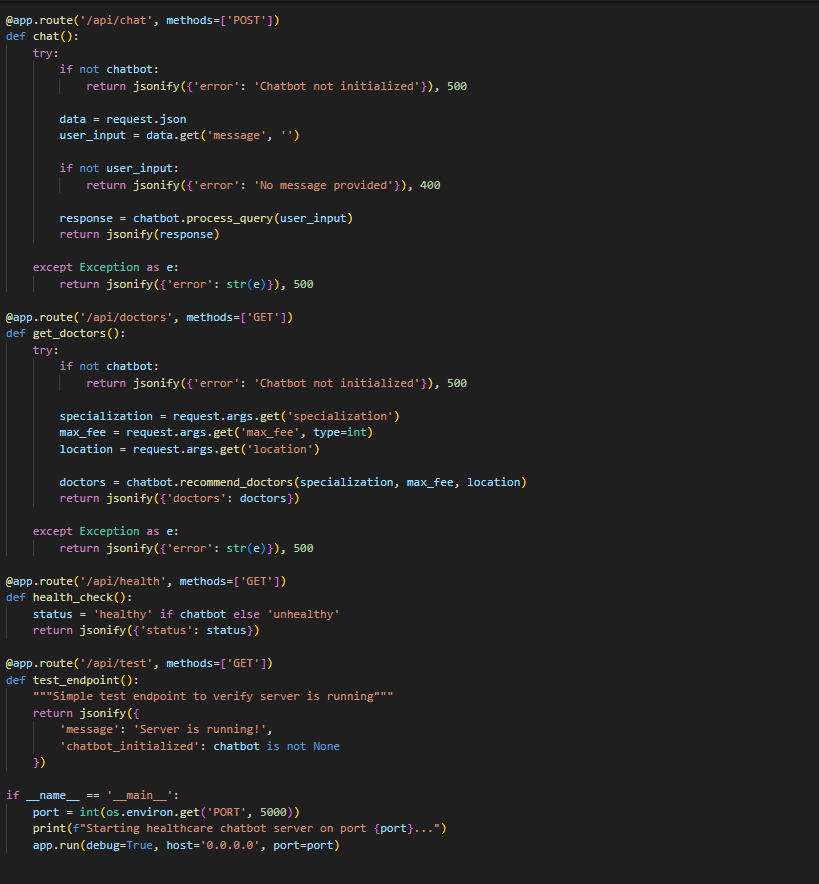
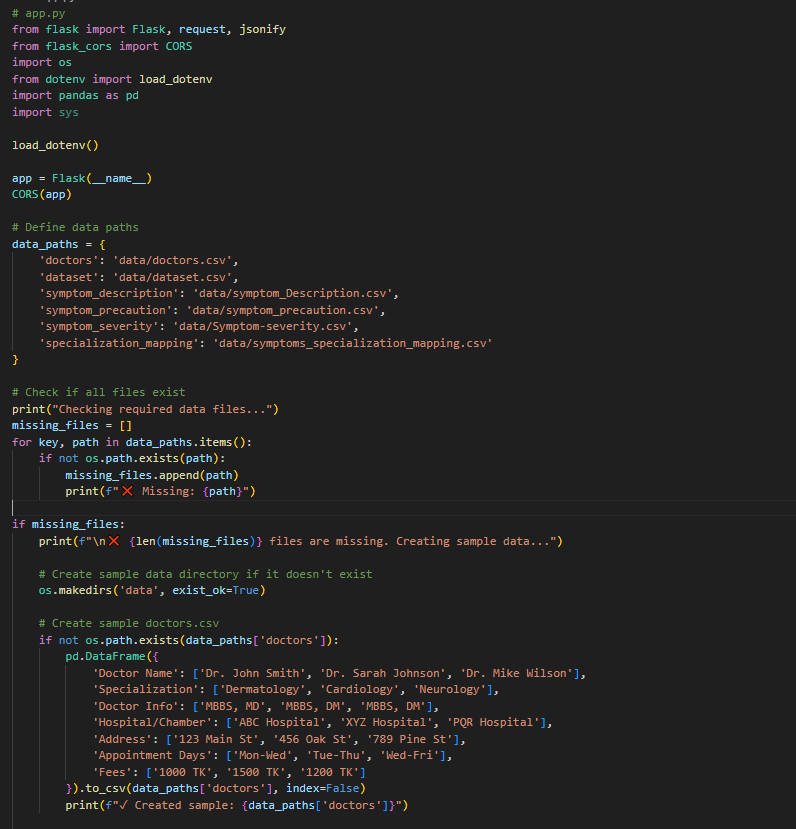
**Figure 4.4: Building the ML Model**

This code is used to build an ML model of the chatbot system that integrates machine learning and rule based methods to analyze medical symptoms and recommend appropriate doctors. The system consists of two main classes: SymptomClassifier, which handles disease prediction from symptoms using a Random Forest model, and HealthcareChatbot, which controls the overall user interaction and response generation.

The SymptomClassifier processes symptom data by converting it into a binary feature matrix where each column represents a symptom and each row represents a patient case. It trains a Random Forest classifier on this data to predict diseases from symptom patterns. The class includes methods for making predictions with confidence scores and saving/loading the trained model for future use.

The HealthcareChatbot class serves as the main interface, initializing the data processor and symptom classifier. It analyzes user input through natural language processing, extracting symptoms and determining whether the query relates to symptom analysis or doctor search. For symptom-based queries, it uses either the ML model or fallback keyword matching to identify potential diseases, then retrieves relevant information about those conditions including descriptions and precautions. For doctor searches, it parses the query to identify specialization preferences, budget constraints, and location preferences before filtering and recommending suitable doctors from the dataset.

The system employs a sophisticated query processing pipeline that recognizes medical keywords, handles both symptom descriptions and doctor search requests, and provides structured responses containing disease information, confidence levels, and doctor recommendations. This combination of machine learning and rule-based approaches allows the chatbot to handle various medical inquiry scenarios while maintaining fallback options when model predictions are uncertain.

**4.5 Creating a Flask web server**

**Figure 4.5: Building a flask web server**

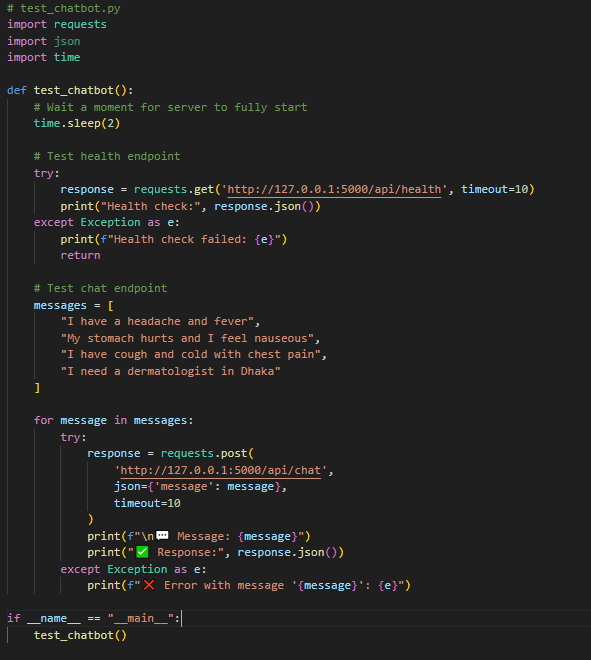
This code creates a Flask web server that acts as the backend API for a healthcare chatbot system. It provides endpoints for users to interact with the symptom analysis and doctor recommendation services through HTTP requests.

The application starts by checking for required medical data files (doctors, symptoms, diseases) and automatically generates sample CSV files if any are missing, ensuring the system can run without immediate data preparation. It then initializes the HealthcareChatbot class from the ML models module, which loads all data and prepares the machine learning models for symptom classification.

The core functionality is exposed through three main API endpoints:

* /api/chat accepts user messages containing symptoms or doctor requests, processes them through the chatbot engine, and returns disease predictions with doctor recommendations
* /api/doctors provides a direct way to search for doctors by specialization, maximum fee, or location
* /api/health and /api/test offer simple status checks to verify the server is running properly

**4.5 Chatbot Performance Checker and Verification**



**Figure 4.6: Code for checking condition of chatbot**

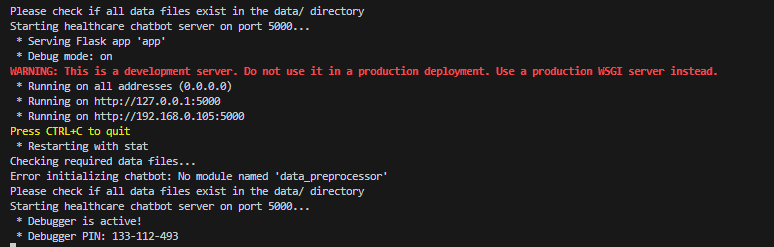
This script tests the healthcare chatbot API by sending sample symptom messages and doctor search requests to the local server. It first checks if the server is healthy, then sends multiple test queries including headaches, stomach pain, and dermatologist searches. The script prints both the user messages and the API's structured JSON responses for verification.



**Figure 4.7: Code for readability of CSV files**

This Python script verifies the existence and readability of CSV data files in the data/ directory. It checks if the directory exists, lists all files with their sizes, and attempts to read each CSV file using pandas. For each valid CSV, it displays the row/column count, column names, and a sample of the data, providing a quick diagnostic of the dataset's structure and integrity.

**4.6 Chatbot Locally Run**



**Figure 4.8: Chatbot initiated on local host**