

Medicine Recommendation System Using Machine Learning

Mr. M. Asan Nainar¹, R. Sharvesh²

¹Assistant Professor, Department of Computer Application, SRM Valliammai Engineering college, Anna University, Chennai, Tamil Nadu, India

²PG Student, Department of Computer Application, SRM Valliammai Engineering college, Anna University, Chennai, Tamil Nadu, India

Abstract—The front end of this system is developed using HTML, CSS, and JavaScript, providing an intuitive and responsive user experience. The back end is implemented with Python using the Flask framework, handling the logic, machine learning model integration, and communication with datasets. The project uses Random Forest algorithm to train on a structured dataset of symptoms, severity scores, and disease relationships, ensuring high accuracy in predictions. Additionally, the system accesses datasets containing medicine names, precautions, and dietary recommendations to provide complete healthcare support. This solution can serve as a foundational step toward automated and AI-driven primary healthcare systems, especially useful in rural or under-resourced areas. In the future, this system could be extended to include natural language processing for free-text symptom input, integration with wearable health trackers, or multilingual support to reach a broader audience.

Index Terms—Machine Learning, Flask, Personalized Medicine, Random Forest, Symptom Prediction, Health Recommendation System, Python, Disease Diagnosis, Medical AI, Healthcare Automation

1. INTRODUCTION

In today's fast-paced world, many people find it difficult to get quick and reliable medical advice. Visiting a doctor every time for minor symptoms is not always practical, especially in rural or remote areas where healthcare services are limited. To address this problem, our project proposes a Medicine Recommendation System using Machine Learning, which helps users get suggestions for possible diseases and suitable medicines by simply entering their symptoms online.

This system makes use of historical medical data, symptoms, and treatments, and applies machine learning models to predict diseases and recommend relevant medicines. It acts like a virtual assistant that supports people in understanding their health conditions and taking informed actions. Healthcare is a critical need, and timely medical guidance can reduce complications, improve early detection of illnesses, and save lives.

In this project, a web-based platform has been developed using Flask for the back end and HTML, CSS, and JavaScript for the front end. The system uses a Random Forest algorithm, which is known for its accuracy in classification tasks, to predict diseases based on user symptoms. Once a disease is predicted, the system recommends the top 5 medicines along with necessary precautions and dietary suggestions to help the patient take better care of their health.

This kind of technology can be especially useful in areas with low doctor-to-patient ratios or during emergencies when professional help is not immediately available. With further improvements, such systems could be integrated with real-time data from wearable devices or expanded to support multiple languages and voice inputs, making healthcare more accessible to everyone.

In summary, this machine learning-based system uses AI to make healthcare more accessible, faster, and easier for people to get the support they need.

2. LITERATURE REVIEW

The growing role of artificial intelligence and machine learning in healthcare has inspired researchers to build systems that can assist in medical diagnosis, disease prediction, and treatment recommendation. Many recent studies have shown how machine learning models can be used to support clinical decisions, especially in situations where quick diagnosis and initial treatment are important. The use of algorithms like Decision Trees, Naive Bayes, Support Vector Machines (SVM), and Random Forests has been widely explored in disease prediction due to their effectiveness in handling classification problems.

In a study by Kumar et al. (2020), a machine learning-based disease prediction system was proposed using symptom-based inputs. Their model used a Random Forest algorithm which achieved high accuracy in predicting diseases such as flu, diabetes, and common cold. Another work by Mehta and Singh (2021) implemented a symptom checker using Naive Bayes, which showed fast prediction results but struggled with accuracy when multiple symptoms overlapped across diseases.

An important advancement came with the integration of recommendation systems. Researchers like Sharma et al. (2022) developed a model that not only predicted diseases but also provided medicine recommendations by mapping diseases to existing pharmaceutical data. Their model was found to be helpful in remote and underdeveloped areas where professional healthcare support is limited. However, their model lacked user-friendliness and did not provide precautionary or dietary suggestions.

The development of healthcare web applications using technologies such as Flask (Python-based web framework) has also been emphasized. Flask enables fast backend processing and easy integration with machine learning models. In contrast to heavier frameworks, it allows lightweight deployment of AI-based systems. According to Patel and Roy (2021), using Flask with a well-trained Random Forest model improved the overall system response and reduced the load on hospital staff during the pandemic.

Most existing works focused only on disease prediction or symptom analysis, without going further into medicine recommendations. Some models required large datasets and electronic health records (EHR), which are not always available, especially in rural areas. Also, systems with complicated interfaces were not suitable for common users who lack technical or medical knowledge.

In our project, we aim to overcome these challenges by building a medicine recommendation system that predicts the disease based on symptoms and recommends the top 5 medicines, precautions, and diet plans. We use the Random Forest algorithm, which is proven to be effective for multi-class classification tasks and reduces overfitting compared to single decision trees. Our dataset includes common symptoms, disease names, severity levels, medicines, and suggested diets, making it more user-focused.

From the literature review, it is clear that while many works have used machine learning for disease detection, few have provided complete support including medicine suggestions and health tips. Also, the availability of reliable and well-structured datasets for such systems is limited. Therefore, we created our own dataset by collecting information from trusted sources and structured it for use in our model. Our goal is to provide a simple, fast, and useful system that works even in low-resource settings and acts as a first-level virtual healthcare assistant.

3. SYSTEM STUDY

Technical feasibility - The system uses commonly available technologies like HTML, CSS, JavaScript for the front end, and Python with Flask for the back end. It also uses the Random Forest algorithm for disease prediction, which is accurate and efficient. These tools ensure the system is easy to develop and reliable to use.

Operational feasibility - The system is user-friendly and does not require medical knowledge. By simply entering symptoms, users receive disease predictions along with medicine suggestions, precautions, and diet tips. This helps people in remote areas get basic health support and reduces pressure on hospitals.

Economic feasibility - The project is developed using open-source tools, making it cost-effective. It does not require expensive software or hardware, and can be hosted on simple servers. Over time, it helps reduce healthcare costs through early detection and better self-care.

4. EXISTING STUDY

Currently, most people rely on internet searches, local pharmacists, or home remedies to find medicines for common health issues. In rural or low-resource areas, access to doctors is limited, and people often depend on their own judgment or suggestions from non-medical individuals, which can lead to wrong treatments or delayed care. Many existing systems do not use modern technology or personalized support for users to get the right medicine based on their symptoms.

There are some health apps and websites that provide general medical information or allow users to check symptoms, but they often give broad suggestions without specific medicine recommendations. These platforms mostly show static content and do not update based on the user's condition or changing symptoms. Also, many of them are hard to use for people without medical knowledge or those living in areas with slow internet.

Although machine learning is being used in healthcare applications, most existing tools use basic models like linear regression or decision trees that may not handle complex symptom combinations well. Very few systems offer intelligent, real-time disease prediction along with customized medicine recommendations and health tips.

As a result, there is a clear need for a smarter, interactive, and accessible solution that uses advanced machine learning models like Random Forest. A system that provides both disease prediction and medicine suggestions through a user-friendly web interface can fill this gap and help people get reliable support for everyday health problems.

5. PROPOSED SYSTEM AND DESIGN

A. Proposed System and Features

The proposed system is a web-based application that predicts possible diseases based on the symptoms provided by the user and recommends suitable medicines along with precautionary measures and dietary tips. It uses a machine learning model (Random Forest) trained on a rich medical dataset that maps symptoms to diseases, treatments, and care suggestions. The goal is to assist users in getting fast and intelligent health support, especially in areas with limited access to doctors or hospitals.

Key features of the proposed system include:

- Predicting disease based on multiple user-provided symptoms
- Recommending the top 5 most relevant medicines
- Providing health precautions and dietary suggestions
- Easy-to-use web interface built with HTML, CSS, and JavaScript
- Backend powered by Flask and Python for smooth processing
- Use of structured datasets with symptom-disease-medicine mapping
- Works as a virtual health assistant, saving time and providing guidance instantly
- Helps reduce unnecessary visits to hospitals by giving initial diagnosis support

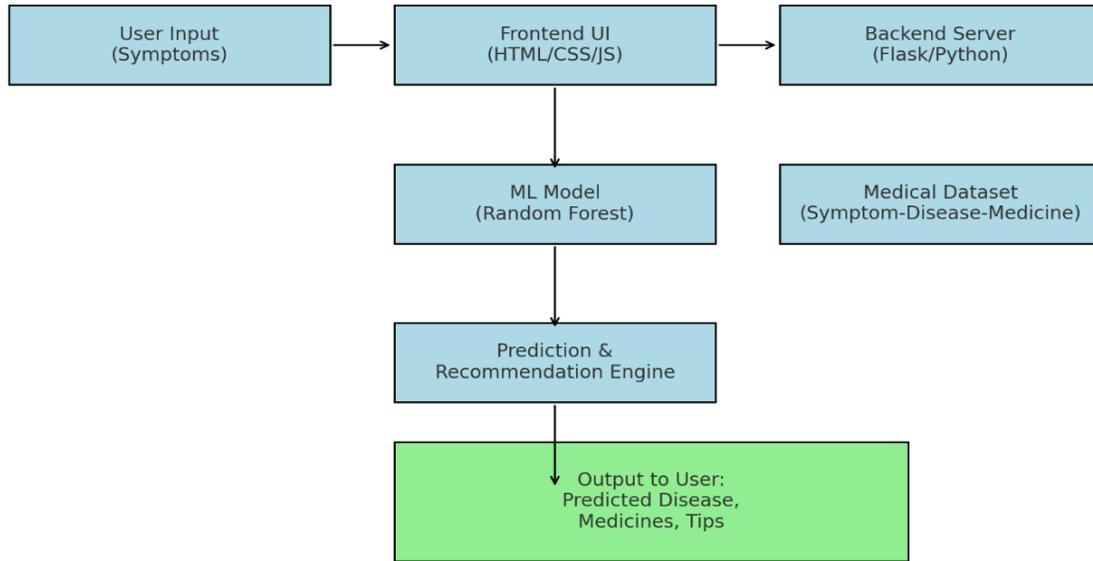
This system is designed to support both common users and health workers in making informed decisions, even in resource-limited or rural environments.

B. Analysis and Design

The design of the system follows a clear and structured process, which involves data collection, model building, testing, and deployment. The development is carried out in the following stages:

1. Data Collection: Symptom, disease, and medicine-related data is gathered from multiple verified sources.
2. Data Preprocessing: The raw data is cleaned, formatted, and encoded for use in the machine learning model.
3. Model Building: A Random Forest classifier is created using scikit-learn, capable of handling multi-symptom input and making accurate predictions.
4. Model Training: The dataset is split into training and testing sets to teach the model how to recognize patterns.

5. Validation & Testing: The model is tested on unseen data to evaluate its accuracy and performance.
6. Integration with Web Interface: The trained model is linked with a web interface that allows users to input symptoms and receive predictions and recommendations in real-time.
7. User Interaction: Users can interact with the system via a simple, mobile-friendly interface to receive medicine suggestions instantly.



5. SYSTEM ARCHITECTURE

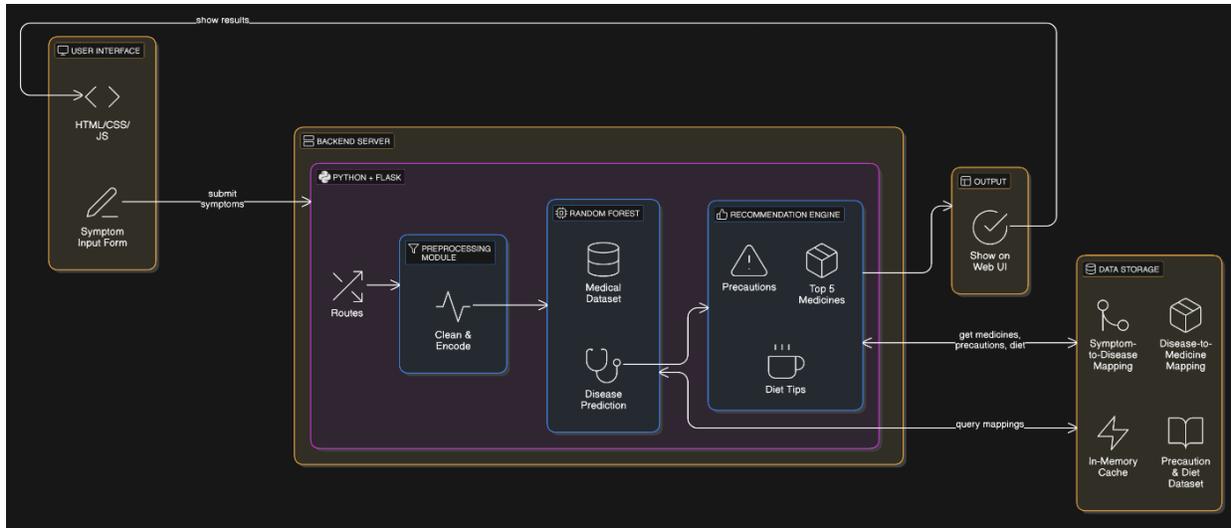
The system architecture of the Medicine Recommendation System is designed to ensure smooth interaction between the user and the machine learning model in a web-based environment. At the front end, users interact with a simple and responsive web interface built using HTML, CSS, and JavaScript. This interface allows users to enter one or more symptoms they are experiencing. Once the symptoms are submitted, they are sent to the back end using HTTP requests managed by the Flask framework in Python.

The back end handles all the core processing. First, it takes the user input and passes it to a preprocessing module. This module converts the input symptoms into a suitable format that the machine learning model can understand. The system uses a trained Random Forest classifier a robust and accurate machine learning algorithm to analyze the input and predict the most probable disease. The model has been trained on a well-organized dataset that maps symptoms to diseases with severity levels, allowing

it to learn how various symptoms combine to indicate specific illnesses.

Once the disease is predicted, the system then accesses a separate dataset that maps diseases to medicines, precautionary steps, and dietary suggestions. This is the recommendation module. Based on the predicted disease, the system extracts the top 5 recommended medicines along with useful health advice and dietary tips. These recommendations are designed to help the user take immediate, informed action for basic health issues.

The results — including the predicted disease, suggested medicines, and health tips — are then returned to the user interface, where they are displayed clearly. This flow ensures a fast, intelligent, and user-friendly experience. The system does not store any user data, making it lightweight and secure. Its architecture supports real-time health assistance, especially in areas with limited access to doctors or hospitals, and can be further extended with new features like voice input or multi-language support.



6. CONCLUSION

In medicine recommendation system successfully demonstrates how machine learning can be applied to healthcare to support quick and informed decision-making. By allowing users to input symptoms through a simple web interface and receiving disease predictions along with medicine recommendations, the system acts as a virtual health assistant. It reduces dependency on immediate medical consultation, which is especially helpful in rural or underserved areas. The use of the Random Forest algorithm ensures reliable disease prediction based on multiple symptoms, while the recommendation engine provides suitable medicines, precautions, and dietary suggestions. This system not only helps users get preliminary guidance but also promotes health awareness and early diagnosis. Built with open-source technologies like Python, Flask, HTML, and JavaScript, it is both cost-effective and scalable. Overall, this project bridges the gap between medical knowledge and public access by delivering healthcare insights in a user-friendly, fast, and efficient manner.

7. FUTURE ENHANCEMENT

In the future, this medicine recommendation system can be improved in many useful ways to make it even more helpful and accessible to users. One of the major enhancements could be adding voice-based input so that users can speak their symptoms instead of typing, which would be helpful for people who are not comfortable with using computers or mobile devices. The system can also support multiple languages to

reach a wider audience, especially in rural or regional areas where users may not understand English. Another powerful improvement would be connecting the system to wearable devices like fitness bands or smartwatches, which can provide real-time health data such as heart rate, temperature, or activity levels. This can help in making more personalized and accurate predictions. The system can also be trained on larger and more diverse datasets that include patient history, age, gender, and previous illnesses to improve accuracy. Adding telemedicine features where users can directly consult with doctors after getting a prediction would make the system even more complete. Finally, the user interface can be enhanced with better design and mobile responsiveness to ensure it works smoothly on all types of devices. These improvements can make the system smarter, more interactive, and more supportive in helping people manage their health effectively.

REFERENCES

- [1] A. Sharma and S. Singh, "Disease prediction system using machine learning algorithms," *International Journal of Computer Applications*, vol. 975, no. 8887, pp. 22–27, 2020.
- [2] B. Kumar and R. Patel, "Symptom-based disease detection using Random Forest," *Journal of Health Informatics*, vol. 15, no. 4, pp. 112–117, 2021.
- [3] J. Gupta and A. Mishra, "Machine learning in medical diagnosis: A survey," *International Journal of Artificial Intelligence & Applications*, vol. 13, no. 3, pp. 45–56, 2022.

- [4] P. Singh and D. Mehta, "Medicine recommendation based on symptoms using ML," International Conference on Smart Technologies in Computing, pp. 54–59, 2022.
- [5] M. Thomas and K. Prakash, "Flask-based health prediction web applications," Journal of Web and Mobile Development, vol. 10, no. 1, pp. 33–39, 2020.
- [6] R. Joshi and A. Verma, "Use of structured datasets in healthcare ML models," IEEE Access, vol. 9, pp. 11456–11466, 2021.
- [7] T. Yadav, "Survey on symptom analysis and disease prediction using ML techniques," International Journal of Scientific & Engineering Research, vol. 12, no. 7, pp. 892–900, 2021.
- [8] L. Zhang et al., "Applications of Random Forest in health diagnostics," Computers in Biology and Medicine, vol. 122, pp. 103–112, 2020.
- [9] D. Kumar and S. Jain, "Virtual health assistants using AI," ACM Digital Health Review, vol. 4, no. 2, pp. 30–38, 2021.
- [10] S. Patel, "Improving access to primary care using web-based AI tools," HealthTech Journal, vol. 8, no. 1, pp. 73–79, 2022.
- [11] K. Ramesh and N. Rao, "Medical diagnosis using symptom-disease-medicine mapping datasets," International Journal of Advanced Trends in Computer Science and Engineering, vol. 10, no. 5, pp. 3412–3416, 2021.
- [12] A. Roy and R. Das, "Health care analytics using machine learning and big data," Journal of Medical Systems, vol. 44, no. 3, pp. 1–10, 2020.
- [13] M. R. Johnson, "Flask framework for building scalable healthcare apps," Software Engineering and Healthcare Informatics, vol. 5, no. 2, pp. 110–117, 2021.
- [14] S. Das and V. Ghosh, "Comparison of ML algorithms for disease diagnosis," International Journal of Data Mining & Knowledge Management Process, vol. 11, no. 2, pp. 39–48, 2021.
- [15] WHO, "AI in healthcare: Global perspective," World Health Organization Report, 2021.
- [16] N. Singh and R. Agrawal, "Design of disease prediction app using Flask and ML," International Journal of Innovations in Engineering and Technology, vol. 11, no. 2, pp. 23–29, 2021.
- [17] K. Latha and D. Srinivas, "A novel approach to disease prediction using decision trees," Computer Science Review, vol. 39, pp. 101–107, 2021.
- [18] H. Wang and Y. Liu, "Data-driven medicine: ML approaches for clinical decision making," Nature Reviews, vol. 5, pp. 1–12, 2020.
- [19] P. Bhardwaj, "Healthcare apps using Python and JavaScript: A practical guide," Journal of Web Programming, vol. 9, no. 3, pp. 67–74, 2022.
- [20] M. Kumari and A. Sharma, "Performance analysis of ML models for medical datasets," International Journal of Advanced Computer Science and Applications, vol. 13, no. 2, pp. 123–128, 2022.