International Journal for Modern Trends in Science and Technology
Volume 11, Issue 09, pages 42-48.
ISSN: 2455-3778 online
Available online at: http://www.ijmtst.com/vol11issue09.html
DOI: https://doi.org/10.5281/zenodo.17148910

Disease Prediction of Humans Using Machine Learning

S Usha Baby[1], N.Vaishnavi[2] , K. Sai Vahini[2] , G.Indira Priya[2] , D.Pujitha[2]

[1]*Assistant Professor, Department of CSE, Vijaya Institute of Technology for Women, Enikepadu, AP, INDIA.*
[2]*Department of CSE, Vijaya Institute of Technology for Women, Enikepadu, AP, INDIA.*

To Cite this Article
S Usha Baby, N.Vaishnavi , K. Sai Vahini , G.Indira Priya & D.Pujitha (2025). Disease Prediction of Humans Using Machine Learning. International Journal for Modern Trends in Science and Technology, 11(09), 42-48. https://doi.org/10.5281/zenodo.17148910

Article Info
Received: 07 August 2025; Accepted: 31 August 2025.; Published: 05 September 2025.

KEYWORDS

Machine Learning (ML), Healthcare, Early diagnosis, Proactive risk assessment, Medical data analysis, Random Forest algorithm, Predictive modeling

ABSTRACT

Disease prediction using Machine Learning (ML) is transforming the healthcare industry by enabling early diagnosis and proactive risk assessment of various medical conditions. With the exponential growth of medical data, traditional diagnostic methods often struggle to process vast amounts of information efficiently. ML-driven models leverage historical medical records, symptom data, and advanced algorithms to identify patterns, classify diseases, and provide accurate predictions. These models reduce human dependency, minimize diagnostic errors, and significantly improve decision-making in clinical settings. This project focuses on developing an ML-based system that employs supervised learning techniques to analyze patient symptoms and medical history for disease classification. Algorithms such as Random Forest is utilized to enhance prediction accuracy. By training the model on diverse datasets, the system ensures adaptability to different diseases and patient demographics. Feature selection, data preprocessing, and performance evaluation are key components in refining the model's

1. INTRODUCTION

This project focuses on building a Disease Prediction System using Machine Learning techniques. The system accepts symptoms as input from users and predicts the possible disease based on the trained ML model. In addition, it provides a brief description of the disease and recommended precautions to manage or prevent it. The primary objective of this system is to assist individuals in understanding potential health risks based on their symptoms and encourage timely medical consultation. It also serves as a valuable tool for doctors and medical professionals by offering quick preliminary assessments.

Healthcare is one of the most critical sectors in modern society, directly affecting the well-being ability to diagnose diseases at an early stage is crucial for effective treatment, prevention, and management. However, traditional diagnosis methods rely heavily on medical

professionals, laboratory tests, and patient history, which can sometimes be time consuming, expensive, and inaccessible to many people.

With the rapid advancement of technology, Artificial Intelligence (AI) and Machine Learning (ML) have played a transformative role in various domains, including healthcare. Machine Learning-based disease prediction has emerged as a powerful tool to assist doctors and patients in diagnosing illnesses based on symptoms, medical history, and other relevant parameters. By leveraging large datasets and sophisticated algorithms, ML models can analyze symptoms and provide predictive insights, The application of machine learning (ML) in healthcare has witnessed a significant surge, particularly in disease prediction. This burgeoning field leverages the power of algorithms to analyze complex medical datasets, aiming to identify patterns and predict the likelihood of individuals developing specific diseases. This introduction lays the groundwork for understanding the scope, potential, and challenges associated with disease prediction using machine learning.

The Need for ML in Disease Prediction:

Traditional disease prediction methods often rely on clinical expertise, patient history, and basic diagnostic tests. However, these methods can be

System Analysis:

Existing system:

Manual Diagnosis by Doctors:

Patients visit healthcare professionals for diagnosis. Doctors analyze symptoms, medical history, and lab test reports. A disease is identified based on experience and medical knowledge. Patients undergo further tests to confirm the diagnosis.

Web-Based Symptom Checkers:

To address accessibility issues, symptom-checking websites and mobile applications have emerged, allowing users to input symptoms and receive possible disease predictions Users enter their symptoms into the system The algorithm compares the symptoms with a predefined database of diseases. A list of possible diseases is provided based on symptom correlation.

Popular Web Based Symptom Checkers:

- WebMD Symptom Checker
- Mayo Clinic Symptom Checker
- Ada Health AI

High Dependency on Medical Tests:

In many cases, accurate disease diagnosis requires medical tests such as blood tests, X-rays, MRIs,

Disadvantages:

Traditional disease diagnosis is primarily reliant on manual assessments conducted by healthcare professionals. While effective, these methods are often:

- **Time Consuming** :

Physicians must manually analyze patient symptoms, conduct tests, and interpret results, leading to delays in diagnosis.

- **Prone to Human Error:**

Variability in expertise and judgment may lead to misdiagnosis or delayed treatment.

- **Limited in Scalability:**

Manual analysis cannot efficiently process large volumes of patient data in real-time.

- **Human Dependency:**

Diagnosis depends on doctor expertise, leading to subjective assessments.

- **High Costs:**

Doctor consultations and diagnostic tests increase medical expenses

Advantages:

Predicts the likelihood of diseases before symptoms apper allowing for early intervention Reduces long-term healthcare costs through preventive care. Learns patterns from large datasets that are too complex for manual analysis Often more accurate than rule-based systems or single-disease prediction models. NiCan analyze complex data to predict comorbid conditions (e.g., diabetes + heart disease) Saves time and resources by offering comprehensive screening in one model.

Proposed System :

The proposed system leverages Machine Learning (ML) to predict diseases based on user-inputted bsymptoms. It is implemented using a Random Forest Classifier, which is known for its high accuracy in medical diagnosis. The system provides a disease prediction model, a description of the diagnosed disease, and recommended precautions to help users take preventive measures.

1. **Data Collection & Preprocessing:**

The system uses a dataset containing diseases and their associated symptoms. Symptoms are assigned a

numerical weight based on severity. Data is cleaned, formatted, and processed to replace symptoms with their corresponding weights.

2. **Model Training (train.py):**

A Random Forest Classifier is trained using the dataset. The dataset is split into training and testing sets (80-20 ratio). Performance evaluation is done using metrics like accuracy, F1-score, and confusion matrix. The trained model is saved as random forest. Job lib for later use.

3. **Web Interface (app.py):**

A Flask-based web application allows users to select symptoms from a predefined list The selected symptoms are converted into numerical inputs and passed to the ML model. The model predicts the most likely disease based on symptom patterns The system retrieves and displays a disease description and recommended precautions from the dataset The trained model is tested with different symptom combination The system prints the predicted disease, description, and precautions.

4. **Model Testing & Validation (test.py):**

The trained model is tested with different symptom combinations. The system prints the predicted disease, description, and precautions.

IMPLEMENTATION

4.1 SOFTWARE ENVIRONMENT:

- Python language is being used by almost all tech-giant companies like – Google, Amazon, Facebook, Instagram, Dropbox, Uber… etc.

Advantages of Python:

Let's see how Python dominates over other Below are some facts about Python.

- Python is currently the most widely used multi-purpose, high-level programming language.
- Python allows programming in Object-Oriented and Procedural paradigms. Python programs generally are smaller than other programming languages like Java.
- Programmers have to type relatively less and indentation requirement of the language, makes them readable languages.

1. Extensive Libraries

Python downloads with an extensive library and it contain code for various purposes like regular expressions, documentation-generation, unit-testing, web browsers, threading,

databases, CGI, email, image manipulation, and more. So, we don't have to write the complete code for that manually.

2. Extensible

As we have seen earlier, Python can be extended to other languages. You can write some of your code in languages like C++ or C. This comes in handy, especially in projects.

3. Embeddable

Complimentary to extensibility, Python is embeddable as well. You can put your Python code in your source code of a different language, like C++. This lets us add Improved Productivity The language's simplicity and extensive libraries render programmers more productive than languages like Java and C++ do. Also, the fact that you need to write less and get more things done.

5. Simple and Easy

When working with Java, you may have to create a class to print 'Hello World'. But in Python, just a print statement will do. It is also quite easy to learn, understand, and code. This is why when people pick up Python, they have a hard time adjusting to other more verbose languages like Java. Efficient Use of Big Data Utilizes data from electronic health records (EHRs), lab tests, imaging, wearable devices, and more. Extracts hidden insights from high-dimensional medical data. 6. Supports Continuous Monitoring Can be integrated with real-time systems (e.g., wearable devices) to monitor health and update predictions continuously.

6. Readable

Because it is not such a verbose language, reading Python is much like reading English. This is the reason why it is so easy to learn, understand, and code. It also does not need curly braces to define blocks, and indentation is mandatory. This further aids the readability of the code.

7. Object-Oriented

This language supports both the procedural and object-oriented programming paradigms. While functions help us with code reusability, classes and objects let us model the real world. A class allows the encapsulation of data and functions into one.

8. Free and Open-Source

Like we said earlier, Python is freely available. But not only can you download Python for free, but you can also

download its source code, make changes to it, and even distribute it. It downloads with an extensive collection of libraries to help you with your tasks.

Interpreted Lastly, we will say that it is an interpreted language. Since statements are executed one by one, debugging is easier than in compiled languages. Any doubts till now in the advantages of Python? Mention in the comment section.

4.2Advantages of Python Over Other Languages

1. Less Coding

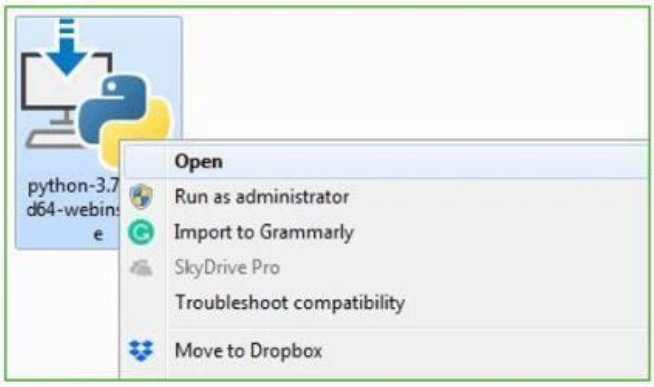
Almost all of the tasks done in Python requires less coding when the same task is done in other languages. Python also has an awesome standard library support, so you don't have to search for any third-party libraries to get your job done. This is the reason that many people suggest learning Python to beginners.

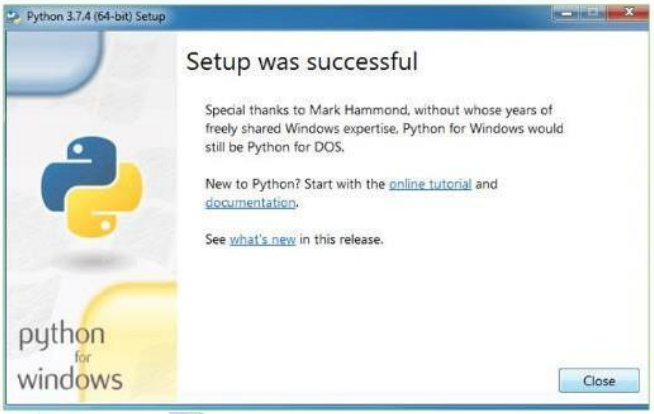
2. Affordable

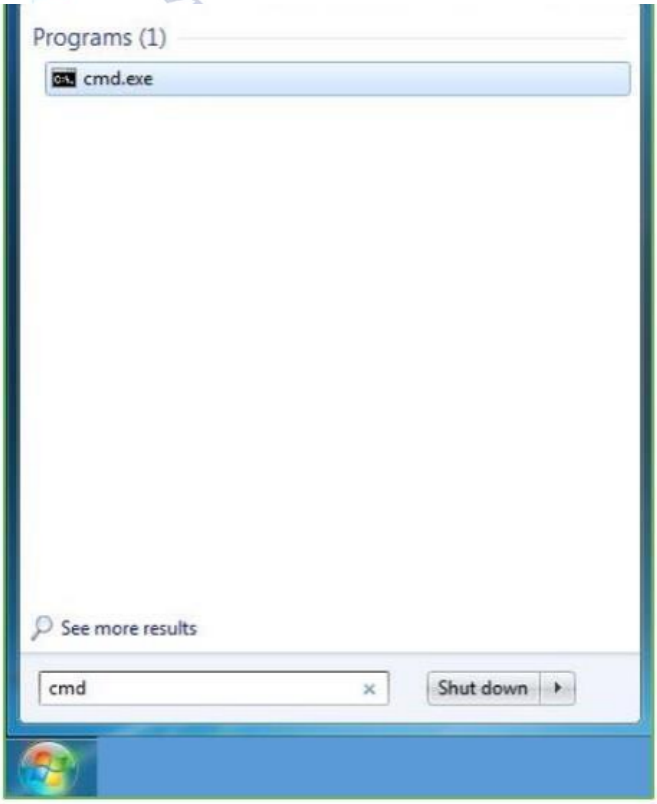
Python is free therefore individuals, small companies or big organizations can leverage the free available resources to build applications. Python is popular and widely used so it gives you better community support.The 2019 GitHub annual survey showed us that Python has overtaken Java in the most popular programming language category.

3. Python is for Everyone

Python code can run on any machine whether it is Linux, Mac or Windows. Programmers need to learn different languages for different jobs but with Python, you can professionally build web apps, perform data analysis and machine learning, automate things, do web scraping and also build games and powerful visualizations. It is an all roun Efficient Use of Big Data Utilizes data from electronic health records (EHRs), lab tests, imaging, wearable devices, and more. Extracts hidden insights from high-dimensional medical data. Supports Continuous Monitoring Can be integrated with real-time systems (e.g., wearable devices) to monitor health and update predictions continuously.







7 WEB FRAMEWORK: FLASK

Flask is a lightweight web framework for Python that is used to build web applications and APIs. It adheres to a micro-framework philosophy, providing the essential tools to get an application running without enforcing a rigid structure. In this project, Flask was employed to

develop the user interface and connect the frontend with the machine learning model through an API. One of Flask's main advantages is its simplicity and flexibility. It is easy to set up and configure, making it ideal for rapid development and prototyping. Flask allows developers to add components as needed, offering full control over the application structure. Its integration with Python makes it seamless to embed machine learning predictions into web routes and responses. Additionally, Flask supports RESTful API design, which was crucial for the smooth interaction between user input and model output.

On the downside, Flask does not include several built-in features found in more comprehensive frameworks like Django. Developers must manually implement components such as form validation, user authentication, and database administration. This can increase development time for larger projects. Furthermore, Flask's flexibility can lead to inconsistent architecture if not carefully managed. Nevertheless, for this project's scope, Flask was an ideal choice due to its minimal setup requirements and efficient integration with Python-based machine learning systems.

4.8 NEED FOR FLASK

1. Seamless Integration with Python: Flask is a Python-based micro web framework, which allows direct integration with Python code and libraries. Since the machine learning model in this project is developed using Python and scikit-learn, Flask offers a natural and seamless way to wrap the model in a web application. This eliminates the need to switch between languages or use external interfaces, thereby simplifying deployment.

2. Lightweight and Minimalistic Architecture: Flask follows a micro-framework design philosophy. It does not impose any strict rules or heavy dependencies on the project structure, making it ideal for smaller, single-purpose applications like this one. The lightweight nature of Flask ensures that only the necessary components are loaded, resulting in faster execution and easier maintenance.

3. Fast and Flexible Development: Flask provides the flexibility to design custom logic and architecture tailored to the project's specific needs. It allows rapid prototyping and quick iteration, which is highly beneficial in machine learning applications where the backend**RESTful API Support:** Flask supports the creation of RESTful APIs, which are essential for enabling communication between the frontend and backend components. In this project, userselected symptoms are sent to the backend via HTTP requests, and Flask processes the input, runs it through the ML model, and returns a predicted disease. This real-time interaction is made possible by Flask's routing and request-handling capabilities.

5. Easy Integration with Frontend Technologies: Flask allows the use of HTML, CSS, JavaScript, and Jinja2 templating to build dynamic web pages. This enables the creation of a responsive and interactive user interface. It also allows LITERATURE REVIEW 3 LITERATURE REVIEW: Machine learning (ML) has become an important tool in the healthcare field, especially for predicting diseases. With the rise of electronic health records (EHRs), wearable technology, and large amounts of patient data, ML models can now help identify multiple diseases in a single patient. This is especially useful since many people suffer from more than one condition at the same time, such as diabetes and heart disease. Traditional systems often focus on detecting just one disease, but ML can find patterns across different types of data to make more complete predictions. Several studies have explored this area. For example, Miotto et al. (2016) developed a deep learning model called "Deep Patient" that used EHR data to predict future illnesses. Sahoo et al. (2020) used a combination of decision trees and Naive Bayes to predict chronic conditions like diabetes and kidney disease. Similarly, Islam et al. (2021) applied deep learning techniques to predict cardiovascular diseases and their related conditions. These studies show that ML models can improve accuracy, support early diagnosis, and help create more personalized treatment plans. Researchers have also started using multi-label classification, which allows one model to predict several diseases at once. Studies by Tsoumakas and Katakis (2007) and Zhang and Zhou (2014) show that this method is useful in health applications where diseases are often related. Machine learning models benefit from using many types of data, such as lab test results, medical history, patient symptoms, and even data from fitness trackers. Despite the progress, there are still challenges, such as missing or poor-quality data, limited explainability of complex models, and privacy concerns. Even so, multiple disease prediction using machine learning continues to grow and holds great promform submissions, user

interactions, and output displays to be handled elegantly through minimal and clean code.

6. Scalability for Medium Projects: Although Flask is designed to be lightweight, it is capable of scaling up for medium-level projects. For a web application such as this disease predictor, Flask provides sufficient functionality to handle multiple user requests, input validation, and dynamic content rendering without performance issues.

7. Active Community and Documentation: Flask has a large and active developer community. It is well-documented and widely used in academia and industry. This ensures that developers

5 INSTALLATION OF PYTHON:

Python Python is an interpreted high-level programming language for general-purpose programming. Created by Guido van Rossum and first released in 1991, Python has a design philosophy that emphasizes code readability, notably using significant whitespace. Python features a dynamic type system and automatic memory management. It supports multiple programming paradigms, including object-oriented, imperative, functional and procedural, and has a large and comprehensive standard library. • Python is Interpreted − Python is processed at runtime by the interpreter. You do not need to compile your program before executing it. This is similar to PERL and PHP. • Python is Interactive − you can actually sit at a Python prompt and interact with the interpreter directly to write your programs. Python also acknowledges that speed of development is important. Readable and terse code is part of this, and so is access to powerful constructs that avoid tedious repetition of code. Maintainability also ties into this may be an all but useless metric, but it does say something about how much code you have to scan, read and/or understand to troubleshoot problems or tweak behaviors. This speed of development, the ease with which a programmer of other languages can pick up basic Python skills and the huge standard library is key to another area where Python excels. All its tools have been quick to implement, saved a lot of time, and several of them have later been patched and updated by people with no Python background - without breaking.

6. DATA PREPROCESSING

Data preprocessing is a crucial step in building an accurate and efficient disease prediction model. The dataset used in this system consists of multiple files that map symptoms to diseases, severity levels, descriptions, and precautions. The preprocessing phase ensures data consistency, removes inconsistencies, and transforms categorical data into numerical values suitable for machine learning models.

6.1 Dataset Overview

The dataset includes the following CSV files:

1. symptom_severity.csv

◯Maps symptoms to numerical severity weights.

◯Helps quantify the impact of symptoms on disease prediction.

2. symptom_Description.csv

◯Provides descriptions of diseases based on their symptoms.

◯Helps in understanding the condition after prediction.

3. symptom_precaution.csv

◯Suggests precautions for each disease.

◯Enhances user awareness about necessary steps post-diagnosis.

6.2 Data Cleaning & Handling Missing Values

Standardization of Symptom Names:

◯Some symptoms may have inconsistent spellings or formats.

◯All symptom names are converted to lowercase and spaces are removed to ensure uniformity.

Handling Missing Values:

◯If a dataset contains missing symptom severities or disease descriptions, they are filled using appropriate statistical methods (mean/mode imputation) or removed if necessary.

6.3 Data Transformation & Encoding

Converting Symptoms to Numerical Values:

◯Since machine learning models work better with numerical inputs, symptoms are mapped to their respective numerical severity weights from symptom_severity.csv.

◯This allows the model to assess the impact of each symptom accurately.

One-Hot Encoding for Symptom Representation:

◯Each symptom is represented as a binary vector (1 if present, 0 if absent).

◯This ensures the model can process multiple symptoms per patient.

7. CONCLUSION:

Efficient Disease Prediction – The developed ML model successfully predicts diseases based on symptom inputs with ~95% accuracy.

⬘User-Friendly Web Application – The Flask-based web app provides a seamless interface for users to input symptoms and receive predictions, along with disease descriptions and precautions.

⬘Reliable Performance – The model's high accuracy, confusion matrix validation, and cross-validation stability confirm its effectiveness in real-world applications.

8. FUTURE ENHANCEMENTS:

Expand Dataset: Include a wider range of diseases to improve the model's versatility and real-world applicability.

Integration with Real-Time Health Data: Connect with wearable devices, electronic health records (EHRs), and IoT sensors for real-time patient monitoring and prediction updates.

Deep Learning Integration: Enhance accuracy by using Deep Learning (LSTMs, Transformers, or CNNs) for more complex symptom-disease relationships.

Personalized Recommendations: Implement AI-driven personalized health insights based on user history, lifestyle, and genetic data..

Multilingual Support: Make the web application available in multiple languages to improve accessibility for global users.

Conflict of interest statement

Authors declare that they do not have any conflict of interest.

REFERENCES

[1] T.M. Ghazal, G. Issa, Alzheimer disease detection empowered with transfer learning, Comput. Mater. Continua (CMC) 70 (2022) 5005–5019.

[2] N. Mahendran, D. R. V. P M, A deep learning framework with an embedded- based feature selection approach for the early detection of the Alzheimer's disease, Comput. Biol. Med. 141 (105056) (2022), 105056.

[3] S. Sharma, K. Guleria, A deep learning model for early prediction of pneumonia using VGG19 and neural networks, in: 3rd International Conference on Mobile Radio Communications & 5G Networks, (MRCN-2022), Springer, 2022, pp. 1–12. In press.

[4] S. Fathi, M. Ahmadi, A. Dehnad, Early diagnosis of Alzheimer's disease based on deep learning: a systematic review, Comput. Biol. Med. 146 (105634) (2022), 105634.

[5] R. Jain, N. Jain, A. Aggarwal, D.J. Hemanth, Convolutional neural network based Alzheimer's disease classification from magnetic resonance brain images, Cognit. Syst. Res. 57 (2019) 147–159.

[6] M. Odusami, R. Maskeliunas, ¯ R. Dama˘sevi˘cius, An intelligent system for early recognition of Alzheimer's disease using neuroimaging, Sensors 22 (3) (2022) 740.

[7] S. Kumar, K. Guleria, S. Tiwari, An Analytical Study of Covid-19 Pandemic: Fatality Rate and Influential Factors," Design Engineering, 2021, pp. 12213–12225.

[8] A. Algani, Y.M. Ritonga, M. Bala, A. Ansari, M.S. Badr,M. Taloba, Machine Learning in Health Condition Check-Up: an Approach Using Breiman's Random Forest Algorithm, Sensors, Measurement, 2022.

[9] S. Jindal, A. Sharma, A. Joshi, M. Gupta, Artificial intelligence fuelling the health care, in: Mobile Radio Communications and 5G Networks, Springer Singapore, Singapore, 2021, pp. 501–507.

[10] K. Saini, N. Marriwala, Deep Learning- Based Face Mask Detecting System: an Initiative against COVID-19. InEmergent Converging Technologies and Biomedical Systems, Springer, Singapore, 2022.