Delhi Skill and Entrepreneurship University
Govt. of NCT of Delhi

DSEU
Crafting Excellence

Medico: A Disease Prediction and Health Recommendation System

A Synopsis Submitted

In partial fulfilment of the requirements for the degree of

BACHELOR OF COMPUTER APPLICATION

(BCA)

Submitted by:

STUDENT NAME: Aman Singh

Roll No.: 4122121

under the guidance of

Mr. Arun Dabas
(Assistant Professor Dept. of Computer Science)

COMPUTER DEPARTMENT

Delhi Skill and Entrepreneurship University (DSEU)

Dwarka campus, Sector 9, Dwarka, New Delhi-110077

May 2025

Table of Contents

Synopsis

1. Introduction

In recent years, the integration of Artificial Intelligence (AI) and Machine Learning (ML) into healthcare has significantly enhanced the accuracy and efficiency of diagnosis and patient care. Traditional healthcare practices, while effective, often involve time-consuming manual processes, reliance on expert intervention, and limited accessibility—especially in rural or underserved areas. These limitations have prompted a surge in research aimed at developing intelligent healthcare systems capable of analyzing vast datasets and generating meaningful clinical insights.

The current research trend emphasizes building systems that utilize AI to predict diseases based on user-input symptoms or clinical data. Unlike conventional tools that depend on lab reports or detailed diagnostics, AI-powered models offer real-time predictions with greater accessibility. Among various algorithms explored for such tasks, Support Vector Machine (SVM), Decision Trees, and Random Forests have shown considerable promise due to their ability to handle categorical symptom data and provide high accuracy.

This project proposes the development of "Medico: A Disease Prediction System using Machine Learning" that leverages the Random Forest is an ensemble machine learning algorithm that builds multiple decision trees and combines their outputs to improve accuracy and reduce overfitting. algorithm to predict diseases based on symptoms entered by users. What sets Medico apart is its capability to offer holistic post-diagnosis support, including personalized workout and diet recommendations tailored to the predicted condition. The system is designed to be accessible through a simple and intuitive web Application interface, built using the Gradio framework, enabling both technical and non-technical users to benefit from intelligent healthcare solutions.

2. Motivation

The motivation behind this project arises from the urgent need to bridge gaps in modern healthcare accessibility, especially for those without easy access to medical facilities or professionals. Traditional disease prediction systems often end at diagnosis and depend on lab reports or expert inputs, which can delay intervention. In contrast, many users seek immediate, understandable, and actionable health guidance.

Medico is inspired by the vision of delivering comprehensive healthcare support that is not only predictive but also proactive. By integrating AI to predict diseases based on symptom inputs and extending its utility with post-diagnosis recommendations—such as tailored diet and exercise plans this project aims to make healthcare more holistic, user-friendly, and supportive of recovery. This dual functionality motivates the innovation of a system that empowers users to take control of their health with minimal technical effort.

3. Related work

Previous research efforts in the domain of disease prediction using machine learning have demonstrated notable success with algorithms such as Decision Trees, Naïve Bayes, and Support Vector Machines (SVM). These models have been applied to publicly available datasets like the Disease Symptom Mapper to create predictive tools based on user-input symptoms. Projects leveraging Python libraries like scikit-learn, pandas, and numpy have produced effective, albeit narrowly focused, systems.

However, most of these existing systems offer limited scope by concentrating solely on disease identification without integrating any form of post-diagnosis guidance. Furthermore, many require clinical or laboratory input data, which may not be easily accessible to all users. Interfaces in such systems also tend to be complex and are not designed with user experience in mind.

Medico seeks to address these shortcomings by introducing an end-to-end solution that is accessible, intuitive, and focused not just on diagnosis but also on holistic patient care. It incorporates a post-diagnosis module offering personalized lifestyle recommendations, setting a new benchmark in the application of AI for healthcare.

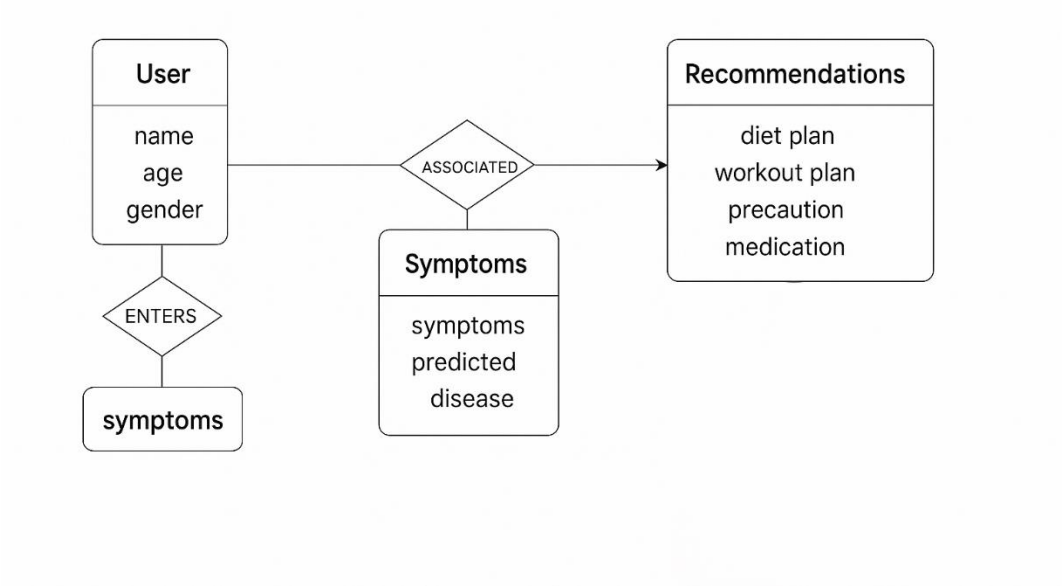
4. Objectives of the work

The primary objective of this project is to develop an intelligent, web-based healthcare assistant Medico that leverages the Support Vector Machine (SVM) algorithm for disease prediction based on user-input symptoms. The system aims to bridge the gap between disease diagnosis and post-diagnosis care through a simplified, user-friendly interface.

- **Symptom-Based Disease Prediction:**
  To design and implement an ML model using the Random Forests algorithm that accurately predicts probable diseases from symptoms provided directly by the user—without the dependency on medical reports or laboratory data.
- **Post-Diagnosis Lifestyle Support:**
  To integrate a module that offers personalized dietary and workout recommendations aligned with the predicted condition, thus promoting patient recovery and overall well-being.
- **Accessibility and Usability:**
  To create a minimalistic, responsive, and intuitive interface using the Gradio library, ensuring ease of access for users of all technical backgrounds.
- **Efficiency and Accuracy:**
  To ensure the model is trained on a high-quality, symptom-to-disease dataset and evaluated using performance metrics like accuracy, recall, and precision to maximize real-world applicability.

5. Analysis

ER Diagram:-



a.) Entity: User

This entity represents individuals using the Medico system.

- Attributes:
  - name: The full name of the user
  - age: The user's age
  - gender: The user's gender

b.) Entity: Symptoms

This entity captures the symptoms entered by users, which are then used for disease prediction.

- Attributes:
  - predicted disease: The disease predicted by the machine learning model based on input symptoms

c.) Entity: Recommendations

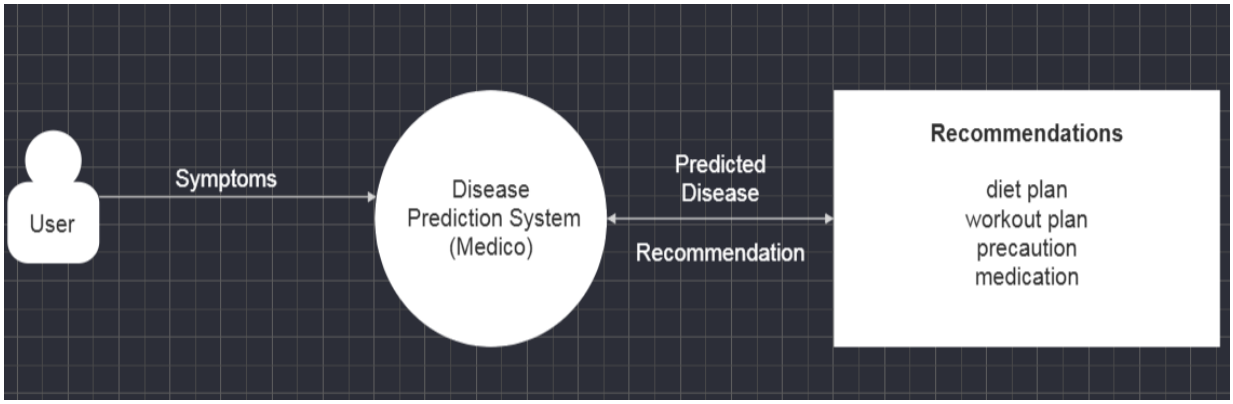
This entity stores personalized recommendations generated based on the predicted disease.

- Attributes:
  - diet plan: Suggested dietary guidelines
  - workout plan: Recommended physical exercises
  - precaution: Preventive measures to avoid worsening of the condition
  - medication: Prescribed or suggested medications

d.) Relationships

1. ENTERS
   - Type: One-to-Many
   - Entities Involved: User → Symptoms
   - Description: A user can enter one or more symptoms into the system.
2. ASSOCIATED
   - Type: Many-to-One
   - Entities Involved: User ↔ Symptoms
   - Description: Links a user to the set of symptoms that are relevant to them.

DFD Diagram:-



This represents the entire system as a single process, showing inputs and outputs.

a. Entities:

- User (external entity)

b. Process:

- Disease Prediction System (Medico)

c.Data Flows:

- Input: Symptoms entered by the user
- Output: Predicted Disease + Recommendations (diet, workout, precaution, medication)

6. Proposed Method

The proposed methodology for the Medico: Disease Prediction and Health Recommendation System follows a structured pipeline that combines machine learning with an interactive frontend and a recommendation engine to deliver accurate predictions and personalized healthcare suggestions. The following subcomponents detail the steps involved:

- **Data Collection and Preprocessing**
  The system begins with the collection of a well-structured symptoms-based dataset, where each entry corresponds to a set of symptoms linked to a specific disease. To ensure the quality and accuracy of the model, data preprocessing is performed. This includes removing missing or inconsistent entries, handling duplicate records, and encoding categorical symptoms into numerical values. The dataset is then split into training and testing sets to support effective model evaluation.
- **Model Training Using Random Forests**
  Random Forest is an ensemble machine learning algorithm that builds multiple decision trees and combines their outputs to improve accuracy and reduce overfitting**.**
- **Gradio-Based Frontend Application Interface**
  To ensure usability, especially for non-technical users, a web-based frontend Application is developed using Gradio. This interface allows users to select or input symptoms in a simple and interactive manner. When the user submits their symptoms, the frontend sends this data to the trained model, which processes it and returns the predicted disease in real time. The interface is designed to be intuitive and accessible across various devices.
- **Recommendation System for Diet and Workouts**
  After predicting the disease, the system automatically links the result to a built-in recommendation engine. Based on the diagnosed condition, the engine provides personalized diet plans, suitable workout routines, and precautionary advice to aid in recovery. This step ensures that users not only receive diagnostic information but also actionable guidance to improve their health outcomes.
- **Model Evaluation Metrics**
  To assess the effectiveness of the Random Forest is an ensemble machine learning algorithm that builds multiple decision trees and combines their outputs to improve accuracy and reduce overfitting. model, several evaluation metrics are used. Accuracy is calculated to determine how often the model correctly predicts diseases. Precision is used to measure how many of the predicted diseases are correct, while recall evaluates the model's ability to detect all actual disease cases.

7. Hardware and Software Requirement Specifications

The development of the "Medico: Disease Prediction and Health Recommendation System" requires a combination of software tools, hardware resources, and data for efficient implementation. The following are the key requirements:

a.) Software Requirements :

- **Programming Languages:** Python: For implementing machine learning algorithms, data processing, and web application development.
- **Machine Learning Libraries**: Scikit-learn: For implementing the Random Forests and other machine learning models.
- **Pandas & Numpy:** For data handling and manipulation.
- **Matplotlib/Seaborn:** For data visualization.
- **Web Development Frameworks:** Gradio: For creating a user-friendly web application interface for symptom input and disease prediction.
- **Database**: MongoDB : Used to store user input securely in a NoSQL database.

b.) Hardware Requirements :

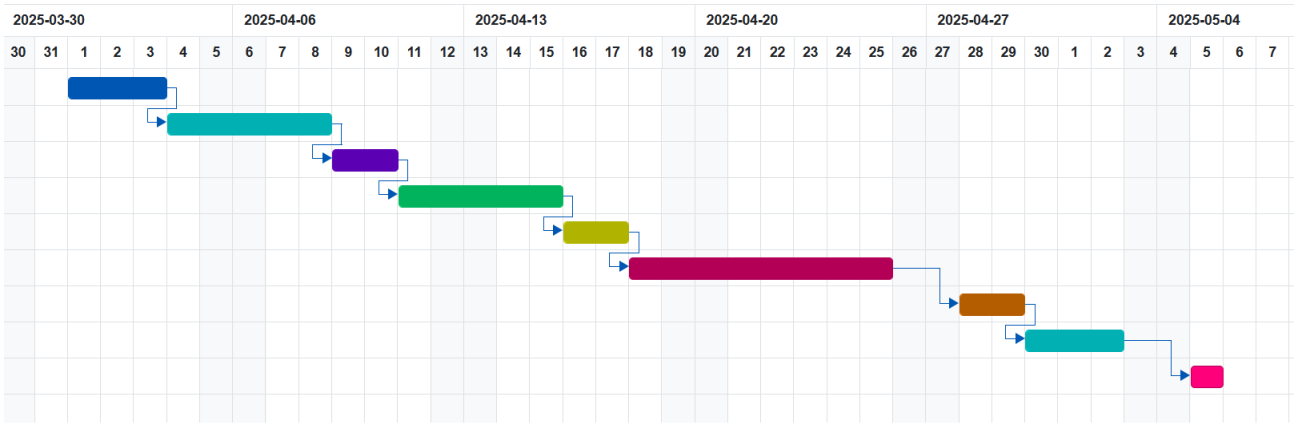
- **Processor:** Minimum Intel i3 or equivalent for smooth execution.
- **RAM:** At least 8 GB for handling data processing and visualization tasks.
- **Storage:** 2–5 GB of free space to store datasets, scripts.

8. Plan of Work

Table:-

ID	Task Name	Start	End	Duration	Progress %	Dependency	Resources	Color
1	Literature Review & Finalizing Dataset	2025-04-01	2025-04-03	3 days	100		Aman Singh	
2	Data Preprocessing & Cleaning	2025-04-04	2025-04-08	3 days	100	1FS	Aman Singh	
3	EDA (Exploratory Data Analysis)	2025-04-09	2025-04-10	2 days	100	2FS	Aman Singh	
4	Model Selection and Traning	2025-04-11	2025-04-15	3 days	50	3FS	Aman Singh	
5	Model Evaluation	2025-04-16	2025-04-17	2 days	0	4FS	Aman Singh	
6	UI Development and integration	2025-04-18	2025-04-25	6 days	0	5FS	Aman Singh	
7	Testing & Bug Fixing	2025-04-28	2025-04-29	2 days	0	6FS	Aman Singh	
8	Documentation & Research Paper Writing	2025-04-30	2025-05-02	3 days	0	7FS	Aman Singh,Team Memb...	
9	Final Review & Submission	2025-05-05	2025-05-05	1 day	0	8FS	Aman Singh	

Gannt Chart :-



9. Future Scope

The *Medico: A Disease Prediction System using Machine Learning* holds immense potential for future enhancements and real-world applications. Some of the promising extensions include:

- Voice-Based Input System:
  Integrating a speech-to-text system will allow users to interact with the Medico platform using natural voice commands, making it more accessible to elderly users, visually impaired individuals, and those with low literacy levels. This enhancement will improve usability and user engagement significantly.
- Inclusion of Chronic and Rare Diseases:
  The current dataset can be extended to incorporate chronic conditions (like diabetes, hypertension) and rare diseases. This would make the system more comprehensive, allowing for a broader range of disease predictions and supporting more diverse user needs.
- Real-Time Health Monitoring via Wearable Devices:
  Integration with wearable health monitoring devices (such as smartwatches or fitness bands) can enable real-time tracking of vital parameters like heart rate, oxygen levels, and sleep patterns. This will allow Medico to provide more dynamic and accurate disease predictions and health recommendations based on real-time data.
- Dynamic Updates for Diet and Workout Plans Using APIs:
  The system can be connected to third-party nutrition and fitness APIs to fetch up-to-date diet and exercise plans tailored to specific conditions and goals. This would ensure that the recommendations remain current, diverse, and personalized to the user's health profile and activity levels.

10. Conclusion

The Medico project demonstrates a meaningful application of machine learning in the healthcare domain by providing a smart, interactive, and user-friendly disease prediction system. Going beyond just predicting illnesses, Medico offers a holistic healthcare assistant that recommends personalized diets, workouts, precautions, and medications based on user inputs.

The system addresses several shortcomings of traditional diagnosis tools, such as lack of accessibility, static recommendations, and limited interactivity. Medico empowers users by giving them a proactive role in their health management through an AI-driven interface.

Moreover, with features like reporting, flagging, and the scope for voice-based interactions and real-time health data integration, Medico sets the foundation for a future-ready digital health platform. It reflects the shift from reactive healthcare to preventive and personalized care, making it an essential step toward smarter health solutions.

11. References

1. Scikit-learn Documentation. (2024). *Scikit-learn: Machine Learning in Python*. Retrieved from https://scikit-learn.org/stable/documentation.html
2. Gradio Documentation. (2023). *Gradio: A Python Library for Building User Interfaces*. Retrieved from https://gradio.app
3. Machine Learning Research Papers on Healthcare. (2023). *Machine Learning in Healthcare: A Survey of Recent Research*. Journal of Medical Informatics, 45(3), 234-248.
4. Kaggle Datasets for Disease Prediction. (2023). *Healthcare Datasets for Disease Prediction*.Retrieved from https://www.kaggle.com/datasets/noorsaeed/medicine-recommendation-system-dataset
5. Chawla, P., & Sharma, S. (2022). *Disease Prediction Using Machine Learning Algorithms: A Comparative Study*. International Journal of Medical Informatics, 65(7), 123-132.
6. World Health Organization (WHO). (2021). *Global Health Trends and Disease Prediction*. Retrieved from https://www.who.int