Project Name: Medicine Recommendation System using Machine Learning

Team Members:

Bajrang Kumar

Ahana Mandal

Sonu Kumar Ranjan

Department: Computer Science & Engineering

Institution Name: Bengal College of Engineering & Technology

Guide Name: Mr. Prasanna Roy (Assistant Professor)

CONTENTS:

Problem Statement

- People often experience health-related symptoms but lack quick access to reliable and accurate medical guidance.
- Searching symptoms online may provide confusing, misleading, or even harmful information.
- Manual symptom checking is time-consuming, error-prone, and not easily accessible for everyone.
- There is a growing need for a **fast, intelligent, and user-friendly system** that can predict diseases and guide users with medical recommendations based on symptoms.

Project Objective

- To develop a **web-based Medicine Recommendation System** that predicts possible diseases based on user-input symptoms.
- To provide **personalized medical guidance** including:
    - Suggested medications
    - Precautionary measures
    - Diet recommendations
    - Workout plans
- To ensure the system is **user-friendly, fast, and accessible**, supporting both text and voice-based symptom input.
- To help users quickly access reliable healthcare information and make informed decisions.

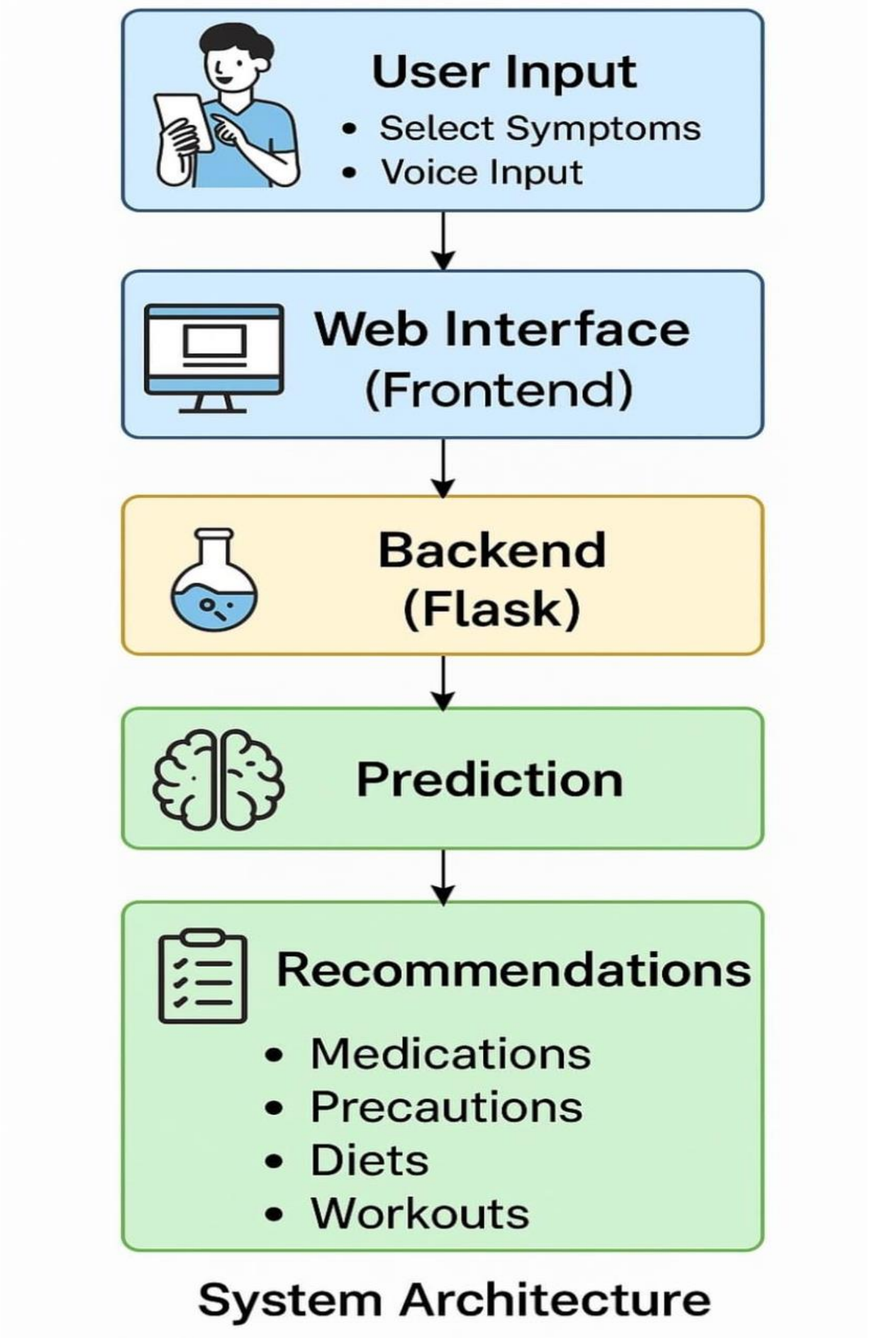
Features of the System

- **Symptom-Based Disease Prediction**: Predicts diseases based on user-input symptoms using machine learning.
- **Speech Input Support:** Allows users to enter symptoms using voice commands for better accessibility.
- **Autocomplete Symptom Selection:** Prevents typing mistakes using a multi-select dropdown with real-time suggestions.
- **Personalized Recommendations:** Provides medications, precautions, diet plans, and workout tips tailored to the predicted disease.
- **User-Friendly Web Interface:** Clean, mobile-friendly, and responsive design for easy navigation.

Technologies Used

- **Frontend:**
  - HTML, CSS, JavaScript
  - Bootstrap for responsive design
  - Select2 for multi-symptom autocomplete dropdown
- **Backend:**
  - Python
  - Flask web framework
- **Machine Learning:**
  - Scikit-learn (Support Vector Classifier, Random Forest, Gradient Boosting, Naïve Bayes)
- **Machine Learning:**
  - Web Speech API for voice-based symptom input
- **Dataset:**
  - Symptom-based disease dataset
  - Supporting files: Precautions, Medications, Diets, Workouts

System Architecture:

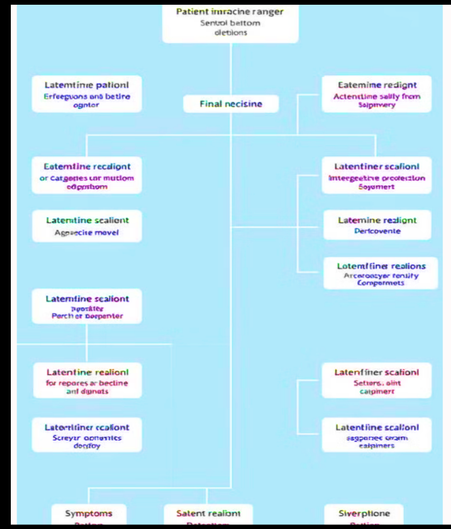


System Architecture

Dataset Details

- **Primary Dataset:** “Training.csv” containing symptom-to-disease mappings.
- **Number of Symptoms:** 132 unique symptoms.
- **Number of Diseases:** 41 classified diseases.
- **Supporting Files:**
  - description.csv: Disease descriptions.
  - precautions_df.csv: Precautionary steps.
  - medications.csv: Suggested medications.
  - diets.csv: Recommended diets.
  - workout_df.csv: Suggested workouts.
- **Dataset:**
  - Symptom-based disease dataset
  - Supporting files: Precautions, Medications, Diets, Workouts

Machine Learning Models Training



Random Forest: Enhanced Accuracy by classifying

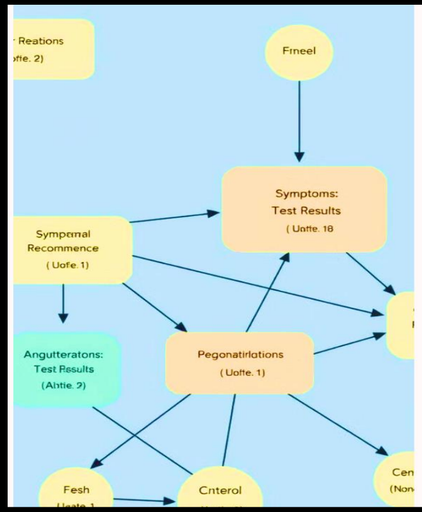
Ensemble Learning

Combines multiple decision trees to make predictions.

Improved Accuracy

Often outperforms linear models, providing more accurate predictions. As we don't predict anything incase of decision trees or random forest . We simply work on the data by bifurcation based on the value of Gini index or entropy

Robust to Noise



Naive Baye's: Probabilistic Insights

1

Probabilistic Approach

Calculates probabilities of different medication recommendations based on patient features.

2

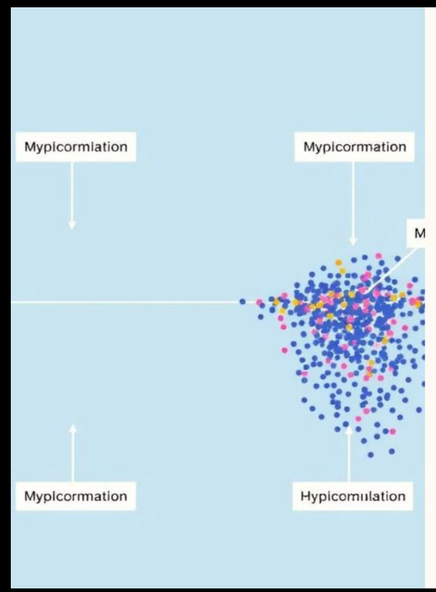
Fast and Efficient

Can make predictions quickly, making it suitable for real-time recommendation systems. It can work well even there is a zero probability of something . Using the alpha based on the values of the classes of each dimension we can do very effective predictions

3

Simple and Effective

Often performs well despite its simplicity, providing valuable insights into medication probabilities.



Support Vector Machine: Seeking Separations

1 **Finding Boundaries**

Identifies a hyperplane that best separates data points into different medication classes.

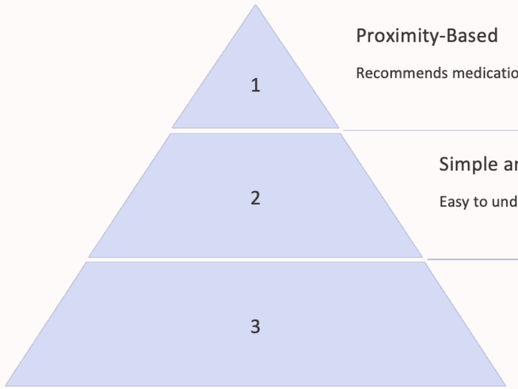
2 **High Dimensional Data**

Model Training and Testing Process with Support Vector Classifier.

3 **Generalisation**

Can generalise well to unseen data, ensuring reliable recommendations.

K-Nearest Neighbours: Finding Similar Cases



Proximity-Based

Recommends medication based on the majority class of its nearest neighbors.

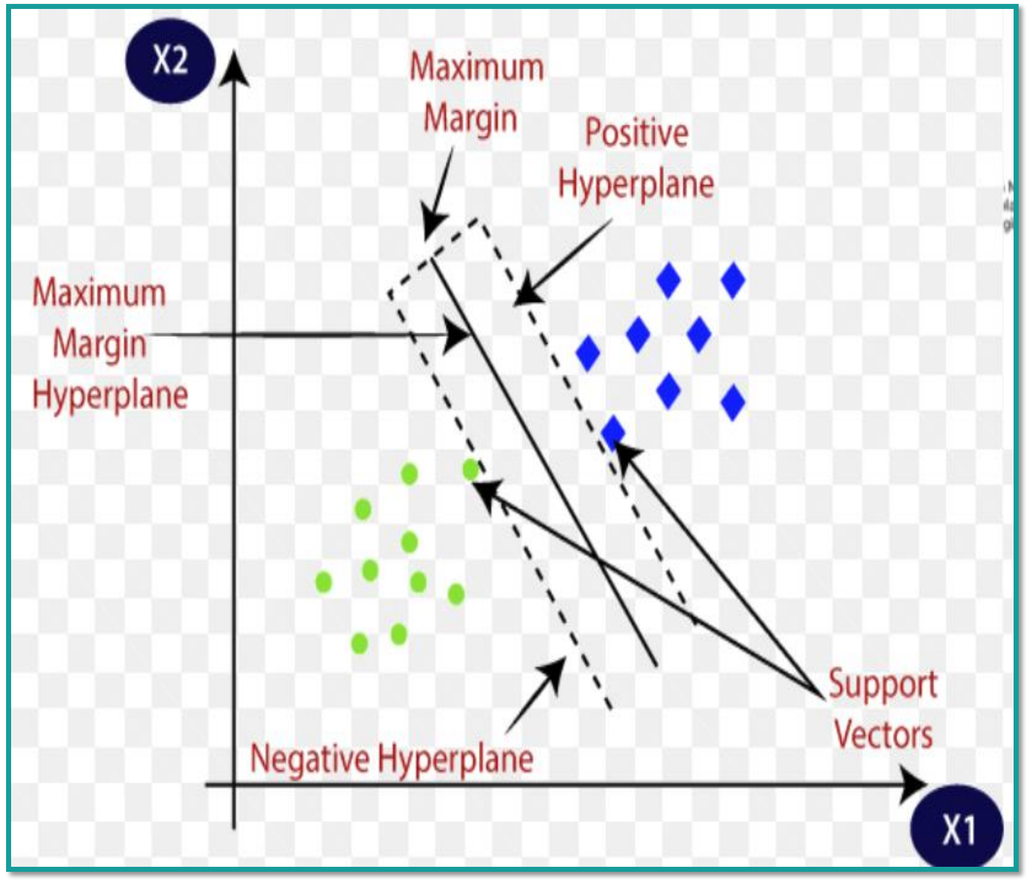
Simple and Intuitive

Easy to understand and implement, making it a good choice for initial testing.

Sensitivity to Data

Performance can be influenced by the quality and distribution of the data.

We selected Support Vector Machine: Seeking Separations



MULTICLASS CLASSIFICATION:- It is simple and effective for Multiclass classification .

FAST TO TRAIN:- It is fast to train and works well for proof of control system

MINIMAL TUNING:- It supported in sklearn with minimal tuning needed.

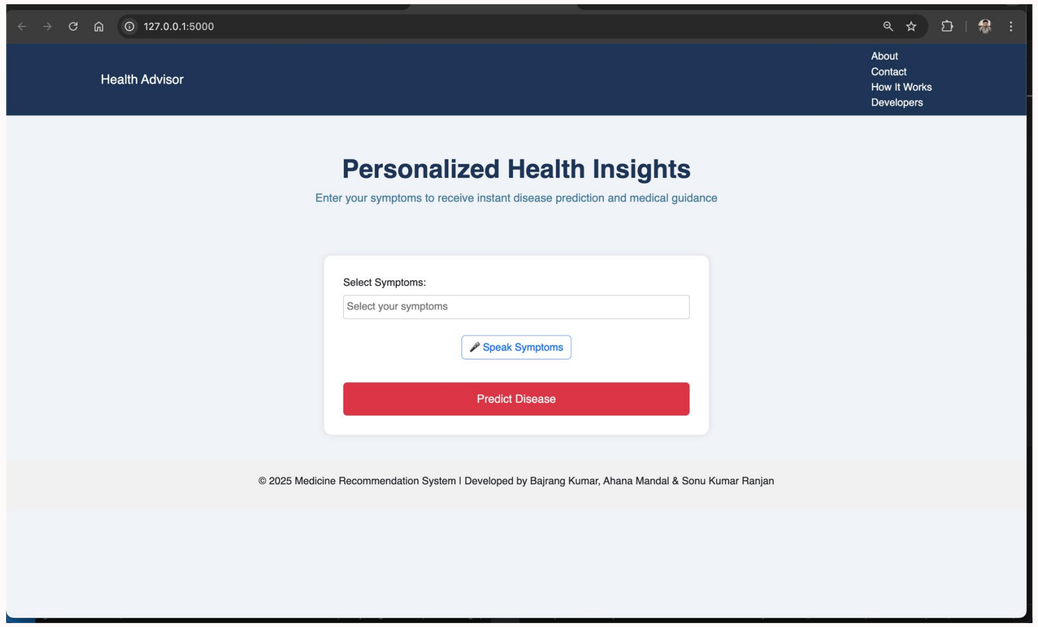
Machine Learning Model

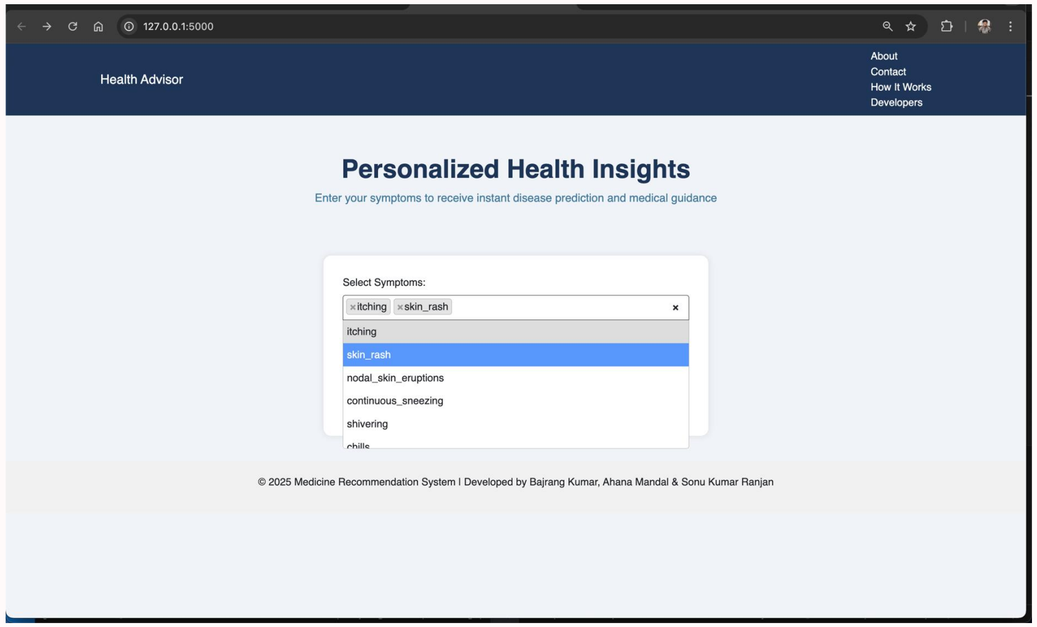
- **Model Used:** Support Vector Classifier (SVC)
- **Why SVC?**
  - Suitable for multi-class classification
  - Simple and efficient for symptom-based datasets
  - Performs well with smaller and structured datasets
- **Input Format:** Binary symptom vector (0 for absent, 1 for present)
- **Training Process:**
  - Dataset split: 70% training, 30% testing
  - Model trained to map symptom patterns to diseases
- **Model Performance:**
  - High accuracy on academic dataset
  - Potential overfitting due to structured dataset (acknowledged for future improvements)

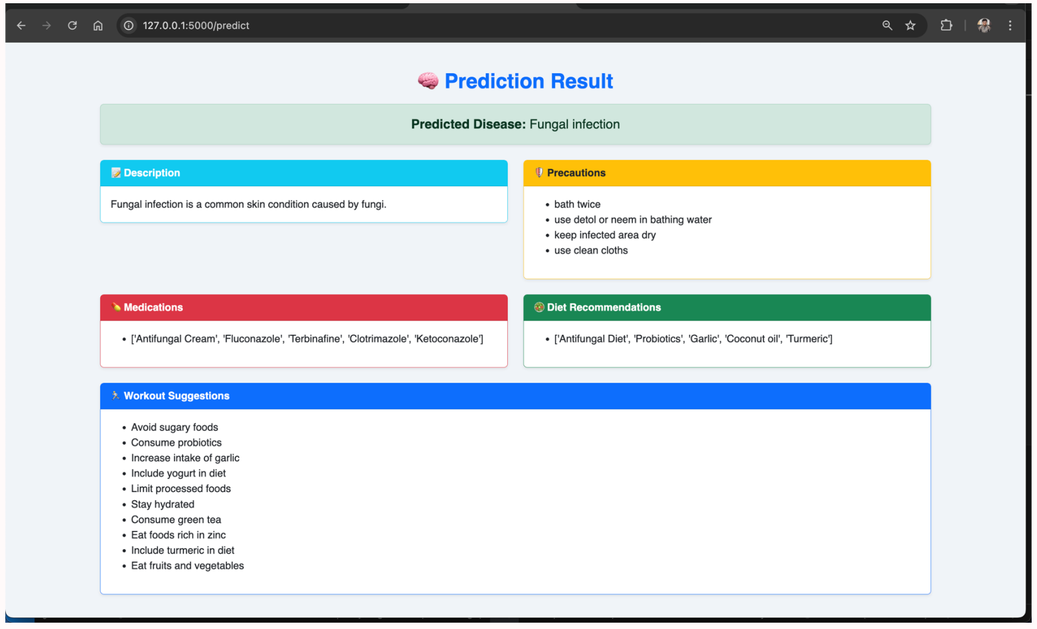
Key Features Explained

- **Symptom-Based Disease Prediction:**
  - Uses machine learning to predict diseases based on multiple symptoms provided by the user.
- **Voice-Based Symptom Input:**
  - Allows users to speak their symptoms directly, making the system easy and accessible for all.
- **Autocomplete Multi-Symptom Selection:**
  - Users can select symptoms using a dropdown with real-time suggestions to avoid input mistakes.
- **Personalized Medical Recommendations:**
  - Displays disease description, suggested medications, precautionary measures, diet plans, and workout tips tailored to the predicted disease.
- **User-Friendly, Mobile-Responsive Interface:**
  - The system is fully responsive and easy to use across devices like laptops, tablets, and smartphones.

Project Demo Screenshots







Challenges Faced

1. Dataset Preparation and Cleaning

- Ensuring correct symptom-to-disease mapping.
- Organizing additional datasets for medications, diets, precautions, and workouts.

2. Symptom Input Validation

- Handling incorrect symptom inputs or misspellings.
- Avoiding system crashes with invalid data.

3. Speech Input Integration

- Managing accurate speech-to-text conversion.
- Ensuring compatibility with the multi-select symptom input.

4. Model Accuracy and Overfitting

- Dealing with possible overfitting due to a small and structured dataset.
- Balancing simplicity and accuracy.

5. User-Friendly Web Design

- Making the system clean, responsive, and mobile-friendly.
- Providing a smooth experience for all users.

Future Enhancements

1. **Improve Prediction Accuracy:**
    - Use more advanced models like XGBoost, or Neural Networks.
    - Expand the dataset with real-world medical records.
2. **Cloud Deployment:**
    - Host the system on cloud platforms like Heroku or AWS for public access.
3. **Mobile Application Developmen:**
    - Build a mobile app version for better accessibility and wider reach.
4. **Multi-Language Support:**
    - Add support for regional and international languages to make the system usable for diverse users.
5. **Real-Time Doctor Consultation:**
    - Integrate real-time doctor suggestions or chatbot support for immediate expert advice.

Conclusion

- Successfully developed a **Medicine Recommendation System** using machine learning.
- Provides **instant disease prediction** based on user-input symptoms.
- Delivers **personalized medical recommendations,** including medications, precautions, diets, and workouts.
- Features a **user-friendly, mobile-responsive web interface** with voice input support.
- Contributes to **faster, accessible, and reliable healthcare guidance.**
- Open for **future improvements** like advanced models, mobile apps, and real-time doctor consultation.

Thank You