Date: 29th April 2024

A COMPREHENSIVE PROJECT REPORT

ON

DISEASE PREDICTION
AND
MEDICAL RECOMMENDATION SYSTEM

Course: CS584 – Machine Learning

Under: Prof. Steve Avsec

BY

Tanmay Pramanick - A20541164

Kunal Rajput - A20540912

Soham Sonar – A20541266

INDEX

I. INTRODUCTION

Welcome to the forefront of personalized healthcare innovation with our project, "Disease Prediction and Medical Recommendation System". This comprehensive system represents a significant leap forward in medical recommendation technology, harnessing the power of machine learning and Python to empower users in understanding and managing their health effectively.

The objective of our project is clear: to develop an end-to-end medical recommendation system that leverages advanced machine learning models to predict diseases based on user-input symptoms. Beyond disease prediction, our system goes a step further by offering personalized medication recommendations, precautionary measures, workout routines, and dietary suggestions tailored to individual health profiles.

II. PROBLEM STATEMENT

In the field of modern healthcare, efficient disease diagnosis and personalized treatment recommendations remain significant challenges. The existing healthcare systems often face limitations in accurately predicting diseases based on patient symptoms and providing tailored medication and lifestyle guidance. This gap not only impacts the quality of patient care but also adds to the burden on healthcare professionals.

This project aims to address this critical issue by leveraging advanced machine learning algorithms to develop a robust medical recommendation system. The primary challenge lies in building a predictive model that can accurately analyze a diverse range of symptoms and associate them with specific diseases, enabling the system to recommend appropriate medications, precautions, and lifestyle modifications. Furthermore, the system must be user-friendly, ensuring seamless interaction and accessibility for individuals seeking reliable healthcare guidance.

III. METHODOLOGY

DATA COLLECTION AND PREPROCESSING

- **Dataset Acquisition:** Obtain the medical dataset comprising symptoms and corresponding diseases.
- **Data Cleaning:** Handle missing values, duplicates, and outliers.
- **Feature Engineering:** Encode categorical symptoms into numerical format using one-hot encoding.

MODEL SELECTION AND TRAINING

Divide the preprocessed dataset into training (70%) and testing (30%) sets.

- **Support Vector Classifier (SVC):**

a) *Mathematical Model:*

$$\text{SVC: } y = f(x) = \text{sign}\left(\textstyle\sum_{i=1}^{n} \alpha_i y_i K(x_i, x) + b\right)$$

where,

- $K(xi,x)$ is the kernel function (e.g., radial basis function),
- αi are the support vector coefficients,
- yi are the class labels, and
- b is the bias term.

b) *Hyperparameters:*

- Kernel type (e.g., radial basis function)
- Regularization parameter CC
- Gamma parameter for non-linear kernels

- **Gradient Boosting:**

a) *Mathematical Model:*

$$F_m(x) = F_{m-1}(x) + \lambda \textstyle\sum_{i=1}^{n} h_m(x_i)$$

where,

- $Fm(x)$ is the boosting model, αi are the support vector coefficients,
- $hm(xi)$ are weak learners (decision trees), and
- λ is the learning rate.

b) Hyperparameters:

- Number of boosting stages (trees)
- Learning Rate

➢ **Random Forest:**

a) Mathematical Model:

$$\text{RF: } y = \text{mode}\left(\text{ensemble}(x)\right)$$

where,

- Ensemble(x) is the set of decision trees.

b) Hyperparameters:

- Number of trees
- Maximum depth of trees

EVALUATION METRICS

➢ **Accuracy:**

$$\text{Accuracy} = \frac{TP+TN}{TP+FP+FN+TN}$$

It measures the proportion of correctly predicted instances (both positive and negative) out of the total instances.

➢ **Precision:**

$$\text{Precision} = \frac{TP}{TP+FP}$$

It quantifies the proportion of correctly predicted positive instances (true positives) out of all predicted positive instances.

➢ **Recall (Sensitivity):**

$$\text{Recall} = \frac{TP}{TP+FN}$$

It measures the proportion of correctly predicted positive instances (true positives) out of all actual positive instances.

- **F1-Score:**

$$\text{F1-Score} = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

F1-Score is the harmonic mean of precision and recall, providing a balance between the two metrics. It considers both false positives and false negatives.

- **Confusion Matrix:**

$$\text{Confusion Matrix} = \begin{bmatrix} TP & FP \\ FN & TN \end{bmatrix}$$

It provides a summary of the performance of a classification model, showing the counts of true positive, false positive, true negative, and false negative predictions.

Here's an explanation of TP, FP, FN and TN:

- **True Positive (TP):**

Definition: Instances that are actually positive (in this case, actual diseases) and are correctly predicted as positive by the model.

Usage: TP represents the number of correct positive predictions made by the model.

- **False Positive (FP):**

Definition: Instances that are actually negative (not diseases) but are incorrectly predicted as positive by the model.

Usage: FP represents the number of incorrect positive predictions made by the model.

- **False Negative (FN):**

Definition: Instances that are actually positive (diseases) but are incorrectly predicted as negative (not diseases) by the model.

Usage: FN represents the number of missed positive predictions by the model.

- **True Negative (TN):**

Definition: Instances that are actually negative (not diseases) and are correctly predicted as negative by the model.

Usage: TN represents the number of correct negative predictions made by the model.

IV. FEATURES AND FUNCTIONALITY

Our medical diagnosis system leverages a comprehensive dataset sourced from Kaggle, incorporating symptoms, diets, workouts, precautions, diseases, medications, and disease descriptions. Users input symptoms using an intuitive frontend interface, and the system predicts:

- **Disease Identification:** Accurate disease prediction based on symptom inputs, providing insights into potential health conditions.
- **Personalized Recommendations:** Customized diets, workout plans, and precautions tailored to predicted diseases, promoting proactive health management.
- **Medication Guidance:** Recommendations for medications based on identified diseases, facilitating informed treatment decisions.
- **Comprehensive Disease Information:** Detailed descriptions of identified diseases to enhance understanding and awareness.
- **Multi-Symptom Analysis:** Supports multiple symptoms input for refined disease prediction, improving accuracy.
- **User-Friendly Interface:** HTML/CSS frontend for seamless user interaction, enabling easy symptom input and prediction retrieval.
- **Interactive and Accessible:** Allows the addition of multiple symptoms to refine predictions, empowering users with actionable health insights.

V. IMPLEMENTATION

a) System Overview:

Our system, the Personalized Medical Recommendation System, is designed to provide users with accurate disease predictions and personalized health recommendations based on their symptoms. This system is user-friendly, powered by advanced machine learning model and integrated into a Flask web application for easy accessibility.

b) Key Features:

- **User-Friendly Interface**

The interface allows users to input symptoms effortlessly, ensuring a seamless user experience.

- **Machine Learning Models**

We've integrated state-of-the-art machine learning algorithms trained on a dataset containing 132 symptoms and 41 prognosis (diseases) to accurately predict diseases based on input symptoms.

- **Tailored Recommendations**

Based on the predicted disease, users receive personalized recommendations:

- ✓ *Medications:* Top 5 recommended medicines and prescription details.
- ✓ *Workout Routines:* Customized workout routines tailored to the predicted disease.
- ✓ *Diets:* Specific dietary recommendations for optimal health management.

- **Flask App Integration**

The entire system is implemented as a Flask web application, ensuring accessibility from any device with a web browser. Users can access healthcare recommendations conveniently, anytime and anywhere.

- **Privacy and Security**

We prioritize user privacy and data security, adhering to the highest industry standards for handling health information.

- **Continuous Improvement**

Our system is designed for continuous improvement. As more data is gathered, our machine learning models evolve, providing increasingly accurate and relevant recommendations.

c) Implementation Details:

➢ *Training Data:*

The system is trained on a dataset (**Training.csv**) containing 132 symptoms and corresponding disease labels (prognosis).

➢ *Additional Data*:

- **medications.csv:** Contains information about medications recommended for different diseases.
- **precautions_df.csv:** Lists precautions to be taken for specific diseases.
- **symptoms_df.csv:** Details symptom-severity relationships.
- **workout_df.csv:** Includes workout recommendations based on diseases.
- **description.csv:** Provides descriptions of different diseases.
- **diets.csv**: Contains dietary recommendations.

➢ *Libraries/Frameworks Used:*

- **pandas:** Used for data manipulation and analysis.
- **scikit-learn (sklearn):** Enables machine learning tasks like model selection, training, and evaluation.
- **flask:** Used to create the web application interface.
- **numpy:** Essential for numerical operations and array manipulations.
- **pickle:** Used for saving and loading trained machine learning models.
- **fuzzywuzzy:** Used for spell checking by selecting symptoms with an 80% or higher similarity score.
- **ast:** Utilized for abstract syntax tree analysis within the project.

These tools collectively facilitate data handling, model development, and web application deployment within the project.

➢ *Machine Learning Model:*

We utilized advanced machine learning models such as Support Vector Classifier (SVC), Gradient Boosting, and Random Forest, which were trained on the symptom-prognosis dataset to accurately predict diseases.

➢ *Flask Application:*

- The Flask web application handles user interactions and prediction requests.
- Symptom input from users is processed by the machine learning model to predict the most likely disease.
- Based on the predicted disease, the application retrieves relevant recommendations from the dataset (medications, precautions, workouts, diets) to display to the user.

➢ *Frontend (HTML/CSS):*

The frontend provides an intuitive user interface for symptom input and displaying recommendations.

VI. Model Training and Evaluation

In our project, we employed a comprehensive approach to model training and evaluation, utilizing Support Vector Classifier (SVC), Gradient Boosting, and Random Forest algorithms. Each of these models was selected based on their suitability for our disease prediction task and their ability to handle complex datasets effectively.

Why SVC, Gradient Boosting, and Random Forest?

- **Support Vector Classifier (SVC):** SVC was chosen for its capability to handle both linear and non-linear classification tasks efficiently. It works well with high-dimensional data, making it suitable for our symptom-prognosis dataset. SVC aims to find the optimal hyperplane that maximizes the margin between different classes, which can lead to robust disease prediction results.
- **Gradient Boosting:** This ensemble learning technique was selected due to its ability to build strong predictive models by combining multiple weak learners. Gradient Boosting iteratively improves the performance of the model by minimizing the errors of previous models, making it suitable for our disease prediction where accuracy is crucial.
- **Random Forest:** Random Forest is a powerful ensemble method that constructs multiple decision trees during training and outputs the mode of the classes as the prediction. It's robust against overfitting and performs well with large datasets, making it a valuable addition to our model selection.

➢ Model Training Process:

- Each model (SVC, Gradient Boosting, and Random Forest) was trained using a labeled dataset consisting of symptoms and corresponding disease labels.
- The training process involved fitting the model to the training data (x_train, y_train), allowing the algorithms to learn patterns and relationships within the dataset.

➢ Model Evaluation:

After training, the models were evaluated using various metrics to assess their performance:

- **Confusion Matrix:** A confusion matrix was generated to visualize the model's performance in terms of true positives, false positives, true negatives, and false negatives.
- **Accuracy:** The accuracy score was calculated to determine the proportion of correct predictions made by the model.
- **Precision, Recall, F1-Score:** These metrics provide insights into the model's ability to make accurate positive predictions (precision), capture all positive instances (recall), and balance between precision and recall (F1-score).

Why Random Forest Specifically?

The decision to adopt Random Forest over Support Vector Machines (SVC) and Gradient Boosting was motivated by its robustness and versatility in handling diverse datasets like the one used in this project. Random Forest is renowned for its ability to mitigate overfitting by aggregating predictions from multiple decision trees trained on different data subsets. This ensemble approach typically leads to better generalization performance on unseen data compared to individual decision trees.

Furthermore, Random Forest excels in handling mixed feature types, such as categorical and numerical data, which is common in medical datasets. Its capability to rank feature importance offers valuable insights into dataset relationships, critical for medical diagnostics. Additionally, Random Forest's flexibility in managing missing data and outliers makes it suitable for real-world datasets with incomplete or noisy information. In summary, the decision to use Random Forest was driven by its adaptability, robustness against overfitting, and effectiveness in handling complex, heterogeneous datasets often encountered in medical applications.

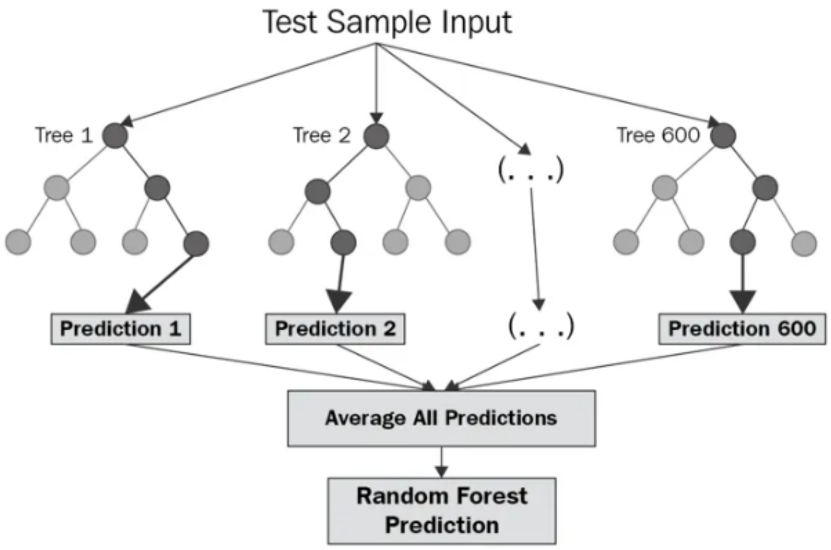


Figure 1: Representation of a Random Forest model

- Saving and Loading Models:

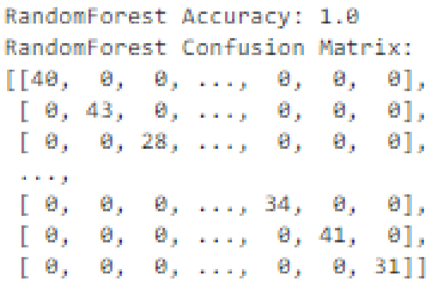
  After training, the Random Forest model was saved using pickle for future use. This allows for easy retrieval and deployment of the trained model without having to retrain it every time.

  This comprehensive approach to model training and evaluation ensures that our disease prediction system delivers accurate and reliable results, leveraging the strengths of different machine learning algorithms. The saved Random Forest model can be easily integrated into our disease prediction pipeline for real-time predictions.

VII. RESULTS

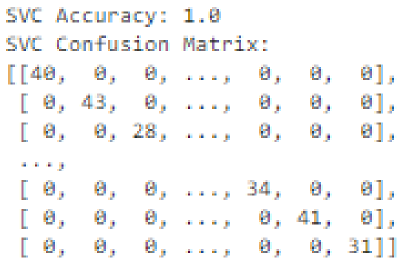
In this section, we assess the performance of three key classifiers: Random Forest, Support Vector Classifier (SVC), and Gradient Boosting. We evaluate each model based on precision, recall, F1-score, accuracy and confusion matrix metrics to gain a comprehensive understanding of their predictive capabilities.

- **Random Forest Classifier:**



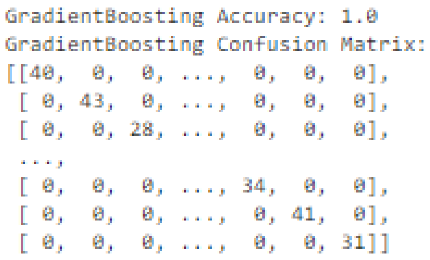
```
RandomForest Accuracy: 1.0
RandomForest Confusion Matrix:
[[40,  0,  0, ...,  0,  0,  0],
 [ 0, 43,  0, ...,  0,  0,  0],
 [ 0,  0, 28, ...,  0,  0,  0],
 ...,
 [ 0,  0,  0, ..., 34,  0,  0],
 [ 0,  0,  0, ...,  0, 41,  0],
 [ 0,  0,  0, ...,  0,  0, 31]]
```

- **Simple Vector Classifier (SVC):**



```
SVC Accuracy: 1.0
SVC Confusion Matrix:
[[40,  0,  0, ...,  0,  0,  0],
 [ 0, 43,  0, ...,  0,  0,  0],
 [ 0,  0, 28, ...,  0,  0,  0],
 ...,
 [ 0,  0,  0, ..., 34,  0,  0],
 [ 0,  0,  0, ...,  0, 41,  0],
 [ 0,  0,  0, ...,  0,  0, 31]]
```

- **Gradient Boosting Classifier:**

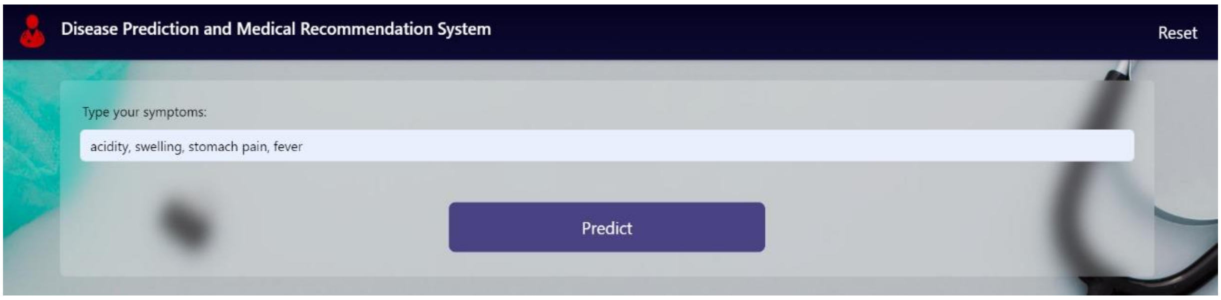


```
GradientBoosting Accuracy: 1.0
GradientBoosting Confusion Matrix:
[[40,  0,  0, ...,  0,  0,  0],
 [ 0, 43,  0, ...,  0,  0,  0],
 [ 0,  0, 28, ...,  0,  0,  0],
 ...,
 [ 0,  0,  0, ..., 34,  0,  0],
 [ 0,  0,  0, ...,  0, 41,  0],
 [ 0,  0,  0, ...,  0,  0, 31]]
```

Precision, recall and F1-score are equivalent for all 3 models.

Features:

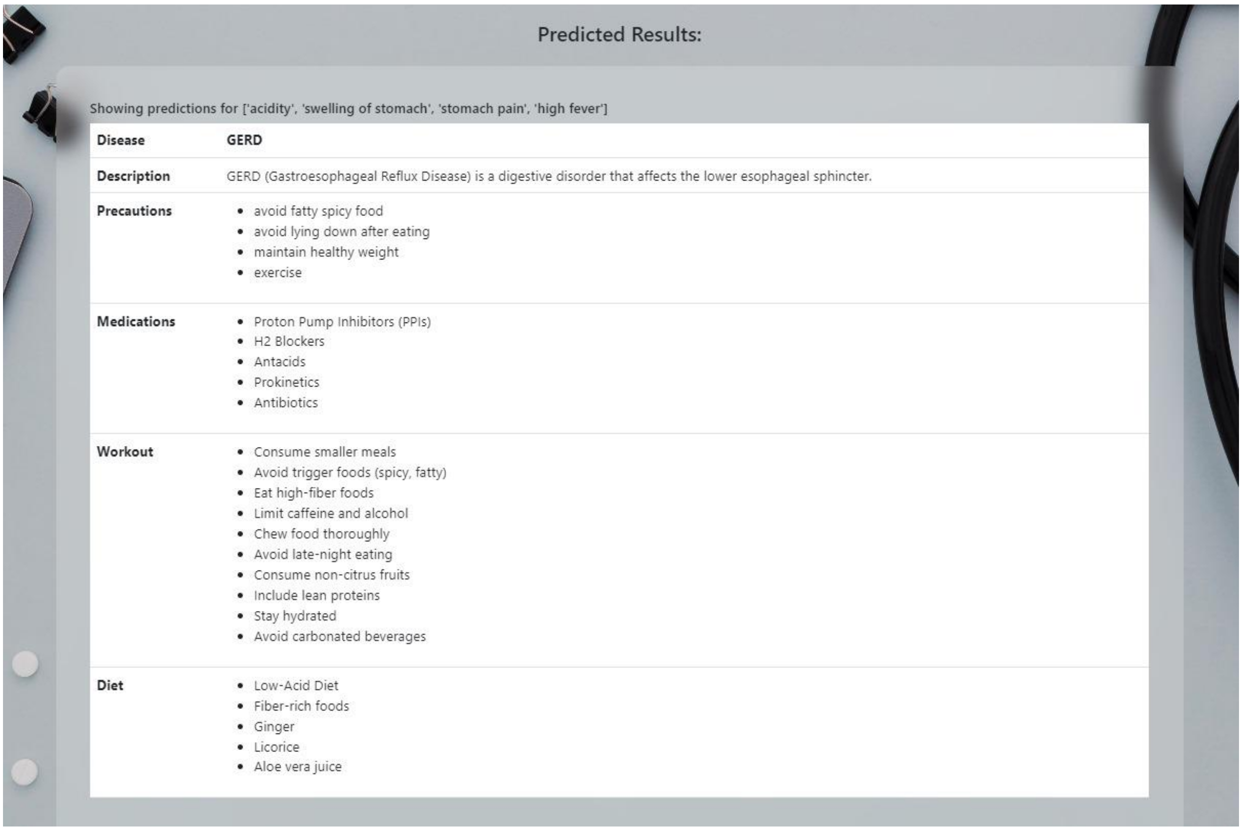
- *Symptoms Input*: Users can enter symptoms to identify potential diseases.
- *Output*: Disease prediction, recommended medication, disease description, diet plan, workout suggestions, and precautions.
- *Interactive Interface:* User-friendly design for seamless interaction.



Type your symptoms:

acidity, swelling, stomach pain, fever

Figure 2: Entering Symptoms to predict the disease

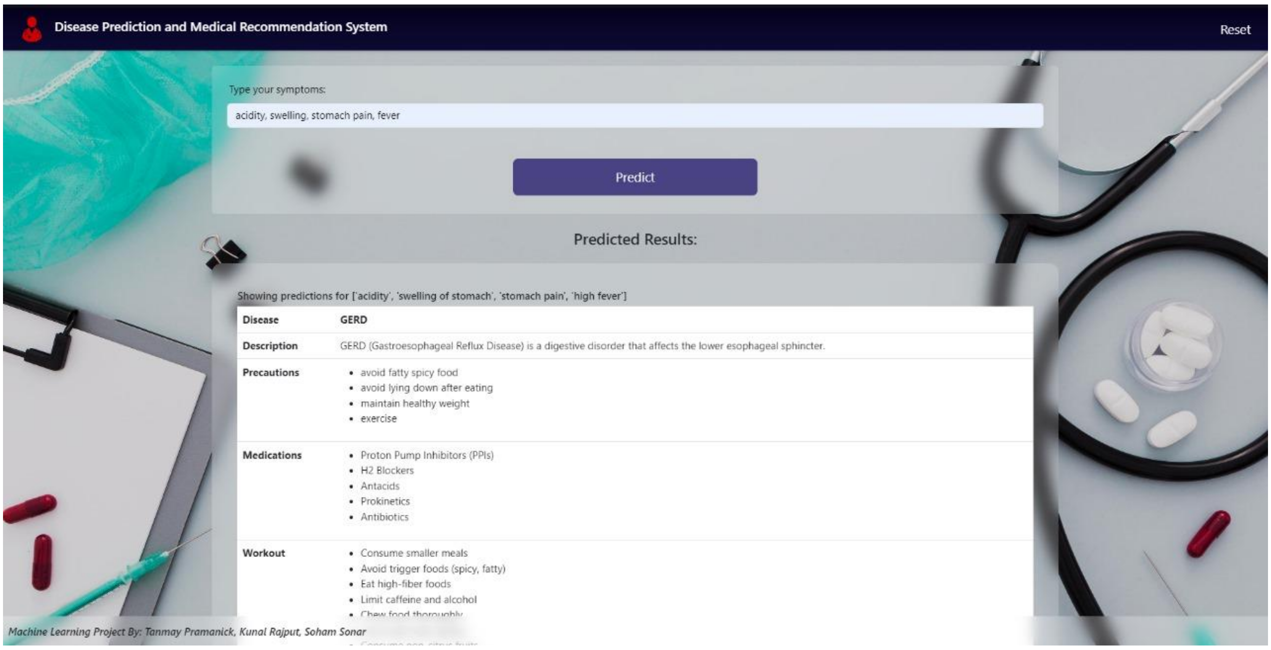


Predicted Results:

Showing predictions for ['acidity', 'swelling of stomach', 'stomach pain', 'high fever']

Disease	GERD
Description	GERD (Gastroesophageal Reflux Disease) is a digestive disorder that affects the lower esophageal sphincter.
Precautions	• avoid fatty spicy food • avoid lying down after eating • maintain healthy weight • exercise
Medications	• Proton Pump Inhibitors (PPIs) • H2 Blockers • Antacids • Prokinetics • Antibiotics
Workout	• Consume smaller meals • Avoid trigger foods (spicy, fatty) • Eat high-fiber foods • Limit caffeine and alcohol • Chew food thoroughly • Avoid late-night eating • Consume non-citrus fruits • Include lean proteins • Stay hydrated • Avoid carbonated beverages
Diet	• Low-Acid Diet • Fiber-rich foods • Ginger • Licorice • Aloe vera juice

Figure 3: Predicted Results of the entered symptoms



Type your symptoms: acidity, swelling, stomach pain, fever

Predicted Results:

Showing predictions for ['acidity', 'swelling of stomach', 'stomach pain', 'high fever']

Disease	GERD
Description	GERD (Gastroesophageal Reflux Disease) is a digestive disorder that affects the lower esophageal sphincter.
Precautions	• avoid fatty spicy food • avoid lying down after eating • maintain healthy weight • exercise
Medications	• Proton Pump Inhibitors (PPIs) • H2 Blockers • Antacids • Prokinetics • Antibiotics
Workout	• Consume smaller meals • Avoid trigger foods (spicy, fatty) • Eat high-fiber foods • Limit caffeine and alcohol • Chew food thoroughly • Consume non-citrus fruits

Machine Learning Project By: Tanmay Pramanick, Kunal Rajput, Soham Sonar

Figure 4: Overall look of our project

VIII. CONTRIBUTIONS

In the development of this project, three team members, Tanmay, Kunal, and Soham, played integral roles in its conception, design, and implementation. Their collaborative efforts resulted in a comprehensive web application aimed at predicting health conditions based on symptoms input by users. Below is a detailed overview of their individual contributions:

- **Tanmay's Contributions:**

Tanmay took charge of the project's technical documentation (report), ensuring clarity and completeness in describing the project's methodologies and outcomes. He also led the data preprocessing phase, meticulously cleaning and transforming the raw dataset to prepare it for modeling. Tanmay was instrumental in model training and selection, experimenting with various machine learning algorithms and ultimately identifying the Random Forest as the most suitable model based on rigorous performance evaluation metrics. His role extended to evaluating the model's generalization ability and robustness through comprehensive testing procedures, ensuring its effectiveness in real-world scenarios.

- **Soham’s Contributions:**

Following Tanmay's work, Soham played a crucial role in model deployment and integration into the project's Flask application. He implemented the trained Random Forest model to enable real-time disease prediction based on user-provided symptoms. Soham engineered custom functions to facilitate model loading, symptom processing, and prediction generation, ensuring efficient and scalable performance within the Flask application. Furthermore, he spearheaded the development of a recommendation system that dynamically provides personalized precautions, medications, workouts, and dietary advice based on predicted diseases, enhancing the application's usability and relevance.

- **Kunal’s Contributions:**

Kunal spearheaded the frontend development efforts, crafting a polished and user-friendly interface using HTML, CSS, and Bootstrap for the Flask application. He architected the main Flask application, orchestrating routing and request handling to seamlessly integrate the frontend with backend functionalities. Kunal's expertise in frontend design was evident in the implementation of the webpage interface, which offers an intuitive platform for users to input symptoms and receive disease predictions. He collaborated closely with team members to synchronize the frontend and backend components, ensuring a cohesive and engaging user experience for the health prediction system.

Together, Tanmay, Soham, and Kunal orchestrated a comprehensive workflow encompassing model development, deployment, and frontend integration to deliver a robust and user-friendly health prediction system. Their collective efforts contributed to the successful implementation of the project, leveraging machine learning for disease identification and providing valuable recommendations to users based on their symptoms.

IX. FUTURE ENHANCEMENTS:

The future potential of this health prediction system is expansive, with several avenues for enhancement. One significant area of advancement could involve integrating more sophisticated machine learning models, such as deep learning architectures, to enhance disease prediction accuracy. Techniques like transfer learning could empower the system to recognize complex patterns and rare diseases by leveraging knowledge from extensive medical datasets. Real-time data integration from medical databases and wearable devices would enable personalized health insights and proactive recommendations based on continuous monitoring.

Furthermore, future enhancements could focus on implementing explainable AI (XAI) techniques to enhance system transparency and interpretability, crucial for building trust in medical applications. Integration with telemedicine platforms could facilitate direct interaction between patients and healthcare professionals based on the system's predictions, enabling early intervention and preventive care. Collaborations with medical research institutions could integrate cutting-edge medical knowledge into the system, ensuring it remains adaptable to emerging healthcare challenges. Ultimately, these advancements could revolutionize disease diagnosis and management, empowering individuals with scalable and accessible healthcare solutions and contributing to the realization of precision medicine.

X. CONCLUSION

In conclusion, the development and implementation of this disease prediction system mark a significant step forward in leveraging machine learning for medical diagnostics. The successful integration of supervised learning models, particularly the Random Forest, has demonstrated promising results in accurately predicting diseases based on input symptoms. Through this project, we have showcased the potential of AI-driven healthcare solutions to augment traditional diagnostic methods, offering accessible and efficient tools for early disease detection and patient care.

While this project has achieved notable success, it's crucial to acknowledge its limitations. The current database size, while sufficient for proof of concept, represents only a fraction of the diverse symptoms and diseases encountered in clinical practice. Scaling up the dataset would address this limitation, enabling the system to handle a broader spectrum of medical conditions with greater precision. However, despite this constraint, the outcomes of this project underscore the transformative potential of AI technologies in reshaping the landscape of medical diagnostics and patient care.

XI. REFERENCES

1) Gupta, J. P., Singh, A., & Kumar, R. K. (2021). A computer-based disease prediction and medicine recommendation system using machine learning approach. Int J Adv Res Eng Technol (IJARET), 12(3), 673-683.

2) Rustam, F., Imtiaz, Z., Mehmood, A., Rupapara, V., Choi, G. S., Din, S., & Ashraf, I. (2022). Automated disease diagnosis and precaution recommender system using supervised machine learning. Multimedia tools and applications, 81(22), 31929-31952.

3) D. Dahiwade, G. Patle and E. Meshram, "Designing Disease Prediction Model Using Machine Learning Approach", 2019 3rd International Conference on Computing Methodologies and Communication (ICCMC), pp. 1211-1215, 2019.

4) M. Patil, V. B. Lobo, P. Puranik, A. Pawaskar, A. Pai and R. Mishra, "A Proposed Model for Lifestyle Disease Prediction Using Support Vector Machine", 2018 9th International Conference on Computing Communication and Networking Technologies (ICCCNT), pp. 1-6, 2018.