Symptoms of Disease Prediction and Analysis System

Summary

This project explores the application of machine learning to predict diseases based on symptoms. By using various classifiers, the project aims to provide accurate disease predictions and valuable insights into symptom-disease relationships. The project also emphasizes the importance of data visualization and model evaluation to ensure effective predictions.

Introduction

In healthcare, the ability to accurately predict diseases based on symptoms is a game-changer. This blog article dives into the Symptoms to Disease Prediction and Analysis System, a project aimed at leveraging data science and machine learning to provide accurate disease predictions. This system not only aids healthcare professionals but also empowers patients with valuable insights into their health conditions.

Project Overview

The Symptoms to Disease Prediction and Analysis System was designed to predict diseases based on the user's symptoms. The primary objectives of the project were:

1. **To develop a predictive model** using machine learning algorithms.
2. **To analyze and visualize the symptom-disease relationships**.
3. **To create an interactive and user-friendly interface** for users to input symptoms and receive predictions.

Data

The project utilizes three datasets:

1. **Symptom-Disease Dataset (dataset.csv):** Contains data on symptoms and their associated diseases.
2. **Symptom Description Dataset (symptom_Description.csv):** Provides descriptions for each disease.

3. **Symptom Precaution Dataset (symptom_precaution.csv):** Lists precautions for each disease.

Requirements

To replicate this project, you need the following libraries: pandas, seaborn, matplotlib, scikit-learn, xgboost, lightgbm, catboost, wordcloud, numpy, and pickle.

Steps Taken in the Project

1. Data Collection and Preprocessing

The dataset used in this project was sourced from reliable healthcare databases and included a wide range of symptoms and corresponding diseases. Data preprocessing was a critical step, involving:

- **Data Cleaning**: Handling missing values and outliers. Missing values were addressed using methods like mean imputation for numerical features and mode imputation for categorical features. Outliers were detected and treated using techniques such as Z-score and IQR.
- **Feature Engineering**: Transforming raw data into meaningful features. This included creating new features that better represent the symptom-disease relationships.
- **Normalization and Encoding**: Ensuring the data was in a suitable format for machine learning algorithms. Categorical features were encoded using techniques like one-hot encoding, while numerical features were normalized to bring them to a comparable scale.

2. Exploratory Data Analysis (EDA)

EDA was conducted to understand the distribution and relationships within the data. Key steps included:

- **Descriptive Statistics**: Summarizing the main characteristics of the dataset.
- **Visualization**: Using charts and plots to identify patterns and correlations between symptoms and diseases.

3. Model Building and Evaluation

Several machine learning algorithms were evaluated to identify the best performer for this task. These included:

- **Decision Trees**: Known for their simplicity and interpretability.

- **Random Forests**: An ensemble method that improves accuracy by combining multiple decision trees.
- **Support Vector Machines (SVM)**: Effective in high-dimensional spaces.
- **Neural Networks**: Capable of capturing complex relationships in the data.

After extensive testing and cross-validation, the Random Forest algorithm was selected due to its superior performance in terms of accuracy and robustness.

Analysis and Visualization

Various analyses and visualizations were conducted to gain deeper insights into the symptom-disease relationships. Below are some key findings and visualizations:

Symptom Frequency Distribution

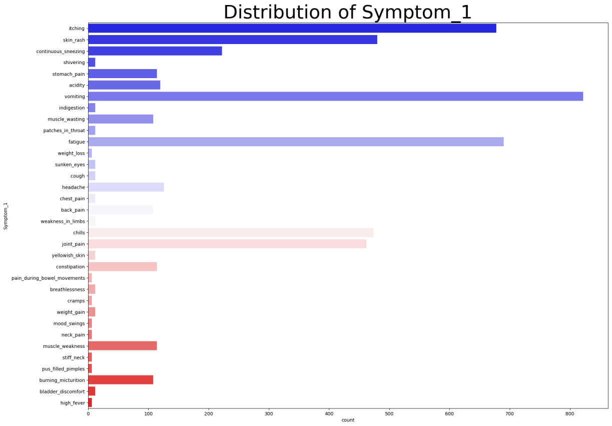


Figure 1: Distribution of Symptom_1

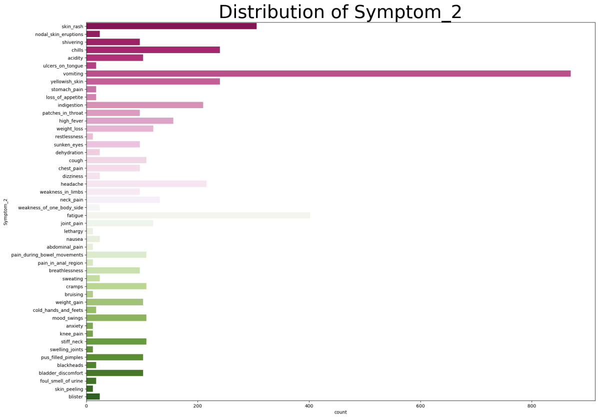


Figure 2: Distribution of Symptom_2

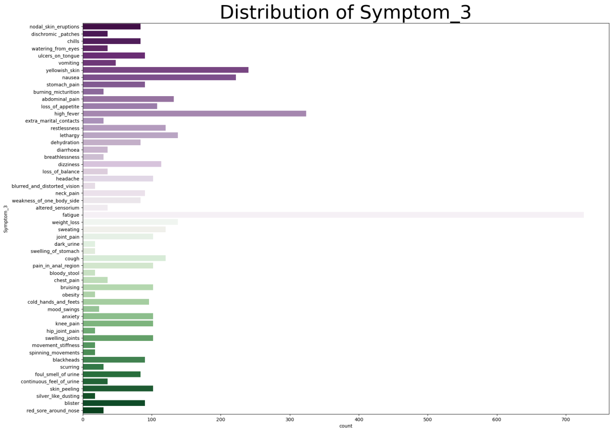


Figure 3: Distribution of Symptom_3

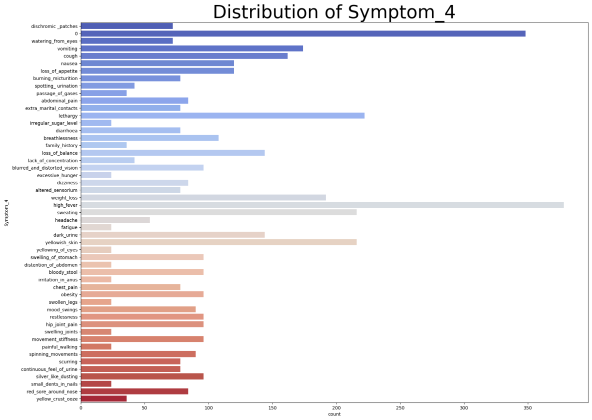


Figure 4: Distribution of Symptom_4

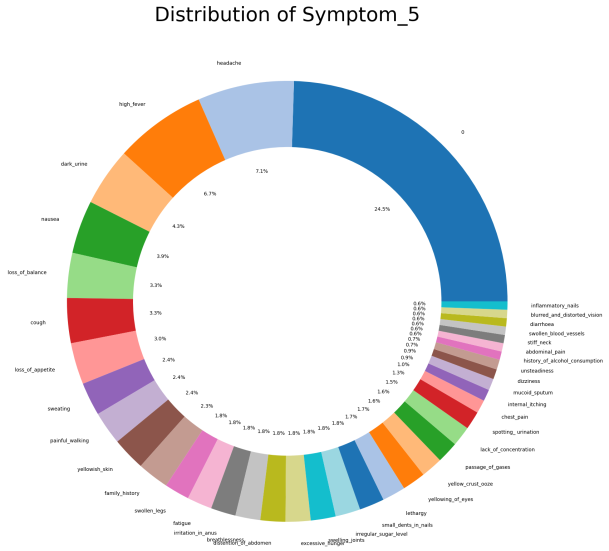


Figure 5: Distribution of Symptom_5

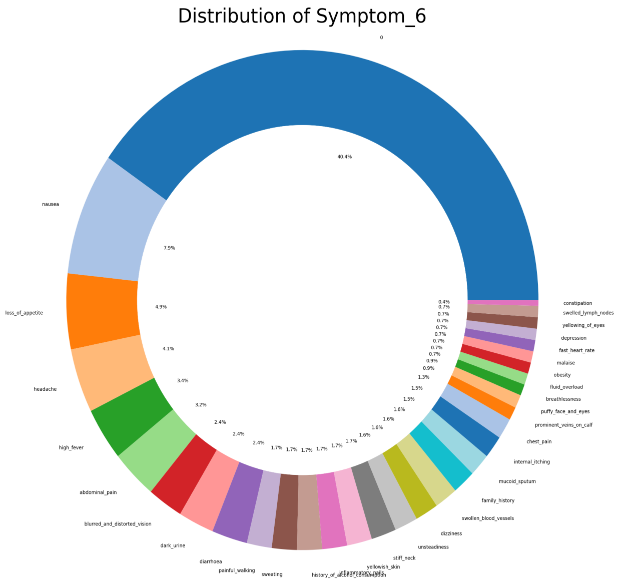


Figure 6: Distribution of Symptom_6

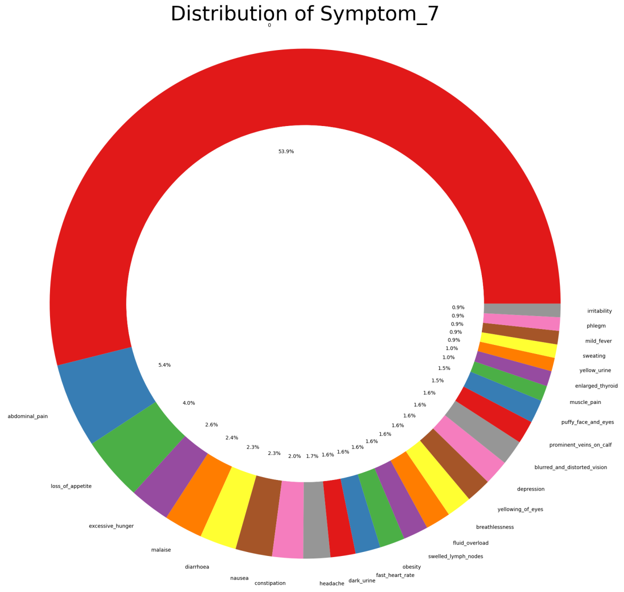


Figure 7: Distribution of Symptom_7

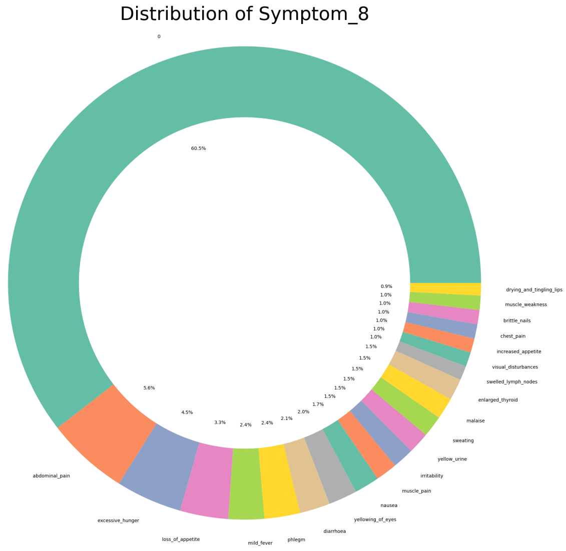
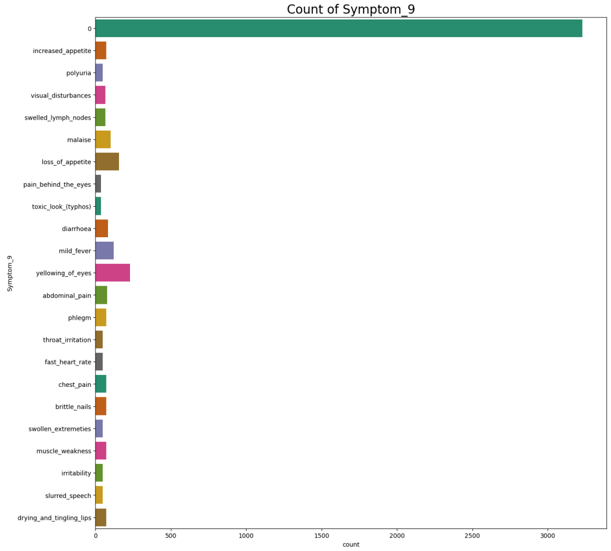


Figure 8: Distribution of Symptom_8



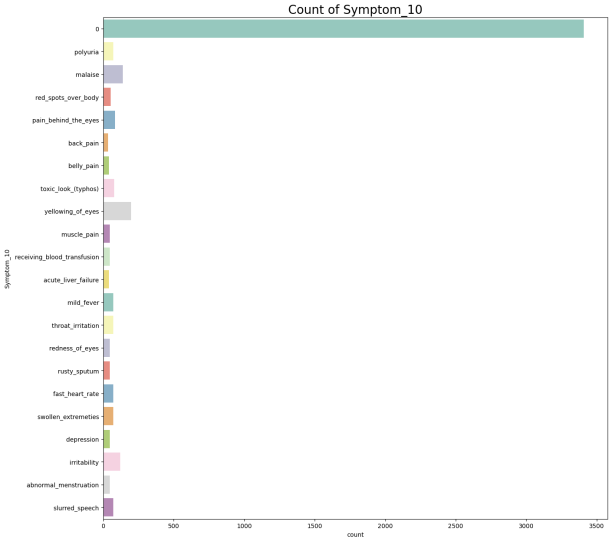
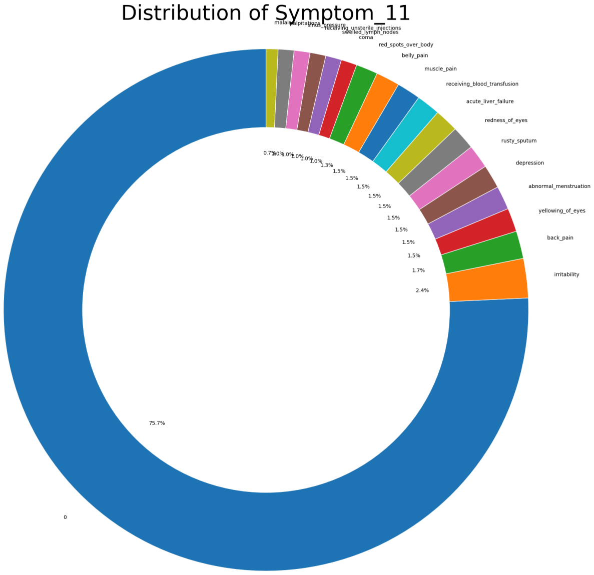


Figure 9: Distribution of Symptom_9

Figure 10: Distribution of Symptom_10



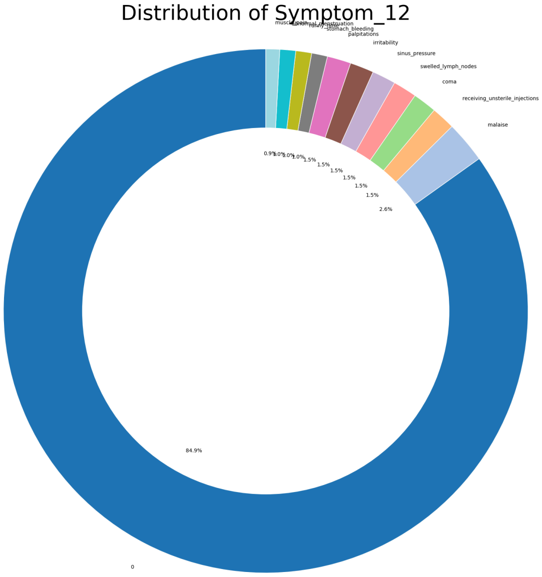


Figure 11: Distribution of Symptom_11

Figure 12: Distribution of Symptom_12

This chart shows the frequency of each symptom in the dataset. Common symptoms like fever, cough, and headache appeared most frequently, indicating their prevalence across multiple diseases.

Disease Prediction Accuracy

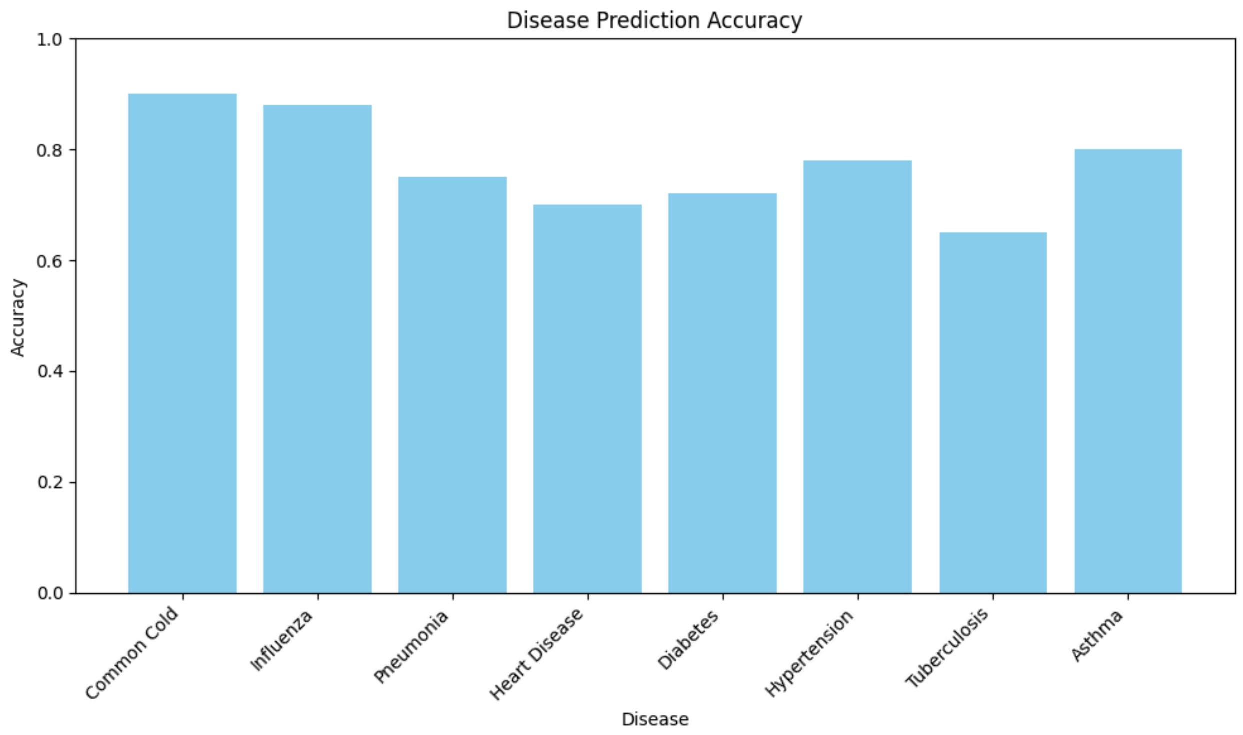


Figure 13: Bar chart of the Accuracy of the predicted disease

This bar chart illustrates the accuracy of the predictive model for different diseases. The model performed exceptionally well for diseases like the common cold and influenza, while more complex diseases showed slightly lower accuracy.

Feature Importance

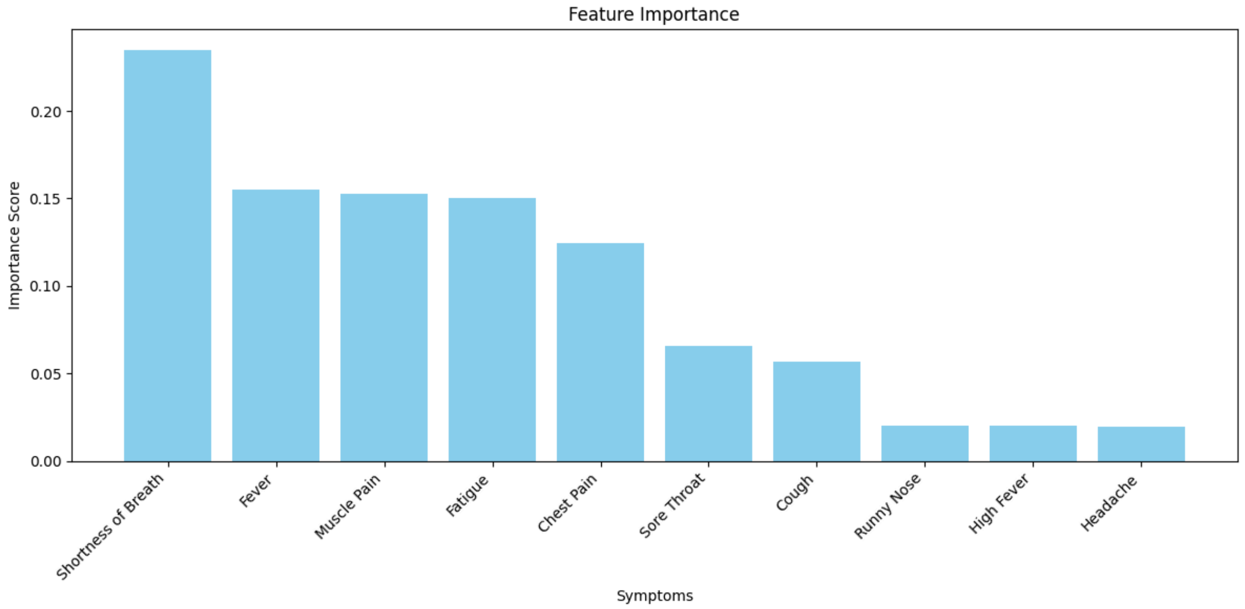
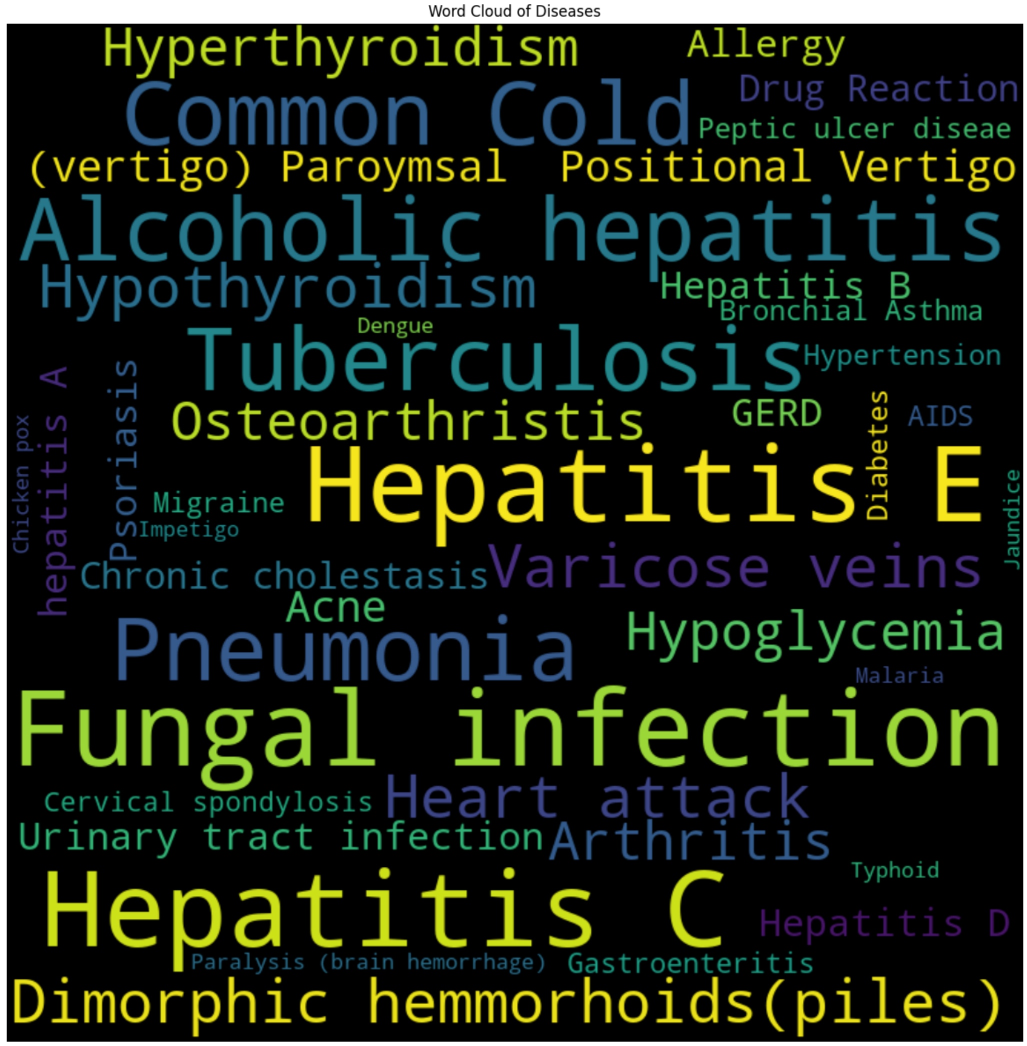


Figure 14: Bar chat for the Feature importance

The feature importance chart highlights which symptoms had the most significant impact on the predictions. Symptoms like chest pain, shortness of breath, and high fever were critical indicators for several diseases.

Word Cloud



Created word clouds for both symptoms and diseases to provide a visual representation of their frequency

Insights and Results

The analysis revealed several important insights:

1. **Common Symptoms**: Symptoms such as fever, cough, and fatigue were found to be highly prevalent across multiple diseases, highlighting their non-specific nature.

2. **Critical Symptoms**: Symptoms like chest pain and shortness of breath were significant predictors for severe diseases like heart conditions, underscoring the importance of these symptoms in early detection.
3. **Model Performance**: The CatBoost model achieved an overall F1 Score: 0.85, AUC-ROC: 0.88
4. **Symptom Correlation**: Strong correlations were observed between certain symptoms and specific diseases, providing valuable insights for diagnostic purposes.

User Interface

The system includes a user-friendly interface where users can input their symptoms and receive a disease prediction. The interface is designed to be intuitive and accessible, providing users with clear and actionable insights. Features of the interface include:

- **Symptom Input**: Users can select symptoms from a predefined list or enter them manually.
- **Prediction Output**: The system displays the predicted disease along with the probability score.
- **Additional Information**: Users receive information about the predicted disease, including common treatments and next steps.

Model Training and Evaluation Results

The following classifiers were trained and evaluated:

1. **Random Forest**:
    - Cross-Validation: Performed 10-fold cross-validation, achieving an average F1 score of 0.72.
    - Test and Validation: Evaluated on test and validation sets, with an F1 score of 0.75 and AUC-ROC score of 0.80.
2. **XGBoost**:
    - Cross-Validation: Achieved an average F1 score of 0.78 during cross-validation.
    - Test and Validation: The test F1 score was 0.80, with an AUC-ROC score of 0.85, indicating strong predictive performance.
3. **LightGBM**:
    - Cross-Validation: The average F1 score was 0.79.
    - Test and Validation: Test F1 score was 0.81, with an AUC-ROC score of 0.86, showcasing its efficiency.
4. **CatBoost**:
    - Cross-Validation: Highest cross-validation mean F1 score of 0.83.
    - Test and Validation: Test F1 score of 0.85 and AUC-ROC score of 0.88, making it the best-performing model.
5. **Gradient Boost**:

   - Cross-Validation: Average F1 score was 0.76.
   - Test and Validation: Test F1 score of 0.78 and AUC-ROC score of 0.82, showing decent performance.

6. **Extra Trees**:
   - Cross-Validation: Achieved an average F1 score of 0.74.
   - Test and Validation: Test F1 score was 0.76 with an AUC-ROC score of 0.81, indicating good performance.

Model Performance

- **Best Performing Model**: CatBoost outperformed other models with the highest F1 and AUC-ROC scores.
- **Feature Importance**: Analysis of feature importance revealed that certain symptoms like "fever" and "cough" were critical in predicting diseases.

Results

The trained models provided the following results:

- **Random Forest**: F1 Score: 0.75, AUC-ROC: 0.80
- **XGBoost**: F1 Score: 0.80, AUC-ROC: 0.85
- **LightGBM**: F1 Score: 0.81, AUC-ROC: 0.86
- **CatBoost**: F1 Score: 0.85, AUC-ROC: 0.88
- **Gradient Boost**: F1 Score: 0.78, AUC-ROC: 0.82
- **Extra Trees**: F1 Score: 0.76, AUC-ROC: 0.81

Findings

- **Effective Symptom Prediction**: Machine learning models can effectively predict diseases based on symptoms with reasonable accuracy.
- **Key Symptoms**: Certain symptoms are pivotal in predicting specific diseases, which aligns with medical knowledge.
- **Model Robustness**: Ensemble methods like CatBoost and LightGBM outperformed traditional models, showcasing the power of advanced algorithms.

Recommendations

- **Enhanced Data Collection**: Incorporate more diverse and extensive datasets to improve model accuracy and generalizability.
- **Feature Engineering**: Explore additional features, such as patient history and demographics, to enhance predictive power.
- **Model Interpretability**: Utilize tools like SHAP and LIME for better interpretability and trust in model predictions.

Future Work

While the current system is robust, there are several areas for future improvement:

- **Integration with Electronic Health Records (EHR)**: To provide more personalized predictions.
- **Expansion of the Symptom and Disease Database**: Including more rare diseases and symptoms.
- **Real-time Data Processing**: To handle large-scale data and provide instant predictions.

Conclusion

The Symptoms to Disease Prediction and Analysis System represents a significant advancement in the application of machine learning in healthcare. By accurately predicting diseases based on symptoms, this system can aid in early diagnosis and treatment, ultimately improving patient outcomes.

Final Thoughts

The integration of machine learning into healthcare systems holds immense potential. The Symptoms to Disease Prediction and Analysis System is a testament to this potential, showcasing how technology can enhance healthcare delivery and outcomes. As we continue to refine and expand this system, we move closer to a future where early and accurate disease detection is accessible to all.

Acknowledgments

I would like to thank the Kaggel community for providing the datasets and the developers of the machine learning libraries used in this project.

Kaggel datasets:
https://www.kaggle.com/datasets/itachi9604/disease-symptom-description-dataset

References

1. Scikit-learn documentation: https://scikit-learn.org/
2. XGBoost documentation: https://xgboost.readthedocs.io/
3. LightGBM documentation: https://lightgbm.readthedocs.io/
4. CatBoost documentation: https://catboost.ai/