International Journal for Multidisciplinary Research (IJFMR)
E-ISSN: 2582-2160 ● Website: www.ijfmr.com ● Email: editor@ijfmr.com

MediBot An AI-Powered Multi-Disease Diagnostic Web Application

Ms.Aruna N[1], Mr.Vedhaprakash Guptha S[2], Mr.Shanmuganandha C S[3], Mr.Thirumurugan M[4]

[1]Assistant Professor, Department of Computer Science and Engineering, Sri Krishna College of Engineering and Technology, Coimbatore, Tamil Nadu, India.
[2,3,4]Student, Department of Computer Science and Engineering, Sri Krishna College of Engineering and Technology, Coimbatore, Tamil Nadu, India,

Abstract:
Medibot is a web-based application that uses artificial intelligence (AI) to predict diseases based on symptoms submitted by users. It uses a modern technology stack and incorporates several machine learning models, including Naive Bayes, Random Forest, Logistic Regression, K-Nearest Neighbors (KNN), and XGBoost. Users can choose from different models and receive predictions along with probabilities for each condition. Medibot emphasizes transparency by providing both global and local explanations for its predictions, helping users understand how their symptoms influence results. The dataset for training the models comes from the Mendeley repository and includes over 246,000 rows of medical data. This paper discusses the architecture, model evaluation, and explanation techniques applied in Medibot, concluding with insights on its effectiveness and potential future improvements in AI-assisted healthcare.

Keywords: AI, Disease Prediction, Machine Learning, Medical Diagnostics, React, FastAPI, Web Scraping, Model Explainability, Logistic Regression, Naive Bayes, XGBoost

1. INTRODUCTION

In recent years, artificial intelligence (AI) has significantly advanced in the healthcare sector. AI models are increasingly used for diagnosis, leveraging large datasets and complex algorithms to assist healthcare professionals in making informed decisions. Medibot is an AI-driven web application that predicts diseases based on symptoms reported by users. It allows users to enter their symptoms, select a machine learning model, and receive predictions about the likelihood of various diseases.

Medibot is designed to help users with technology that can aid in early disease detection, potentially improving treatment outcomes. One of its unique features is the ability to offer multiple machine learning models for disease prediction, ensuring a balance between speed, accuracy, and resource needs. The system also includes clear explanation features, which provide transparency regarding the model's predictions. This transparency is especially crucial in healthcare, where trust in automated systems is vital.

Medibot's architecture uses React for the frontend and FastAPI for the backend. These technologies were selected for their scalability, ease of development, and performance. Additionally, Medibot

regularly updates its dataset and model using Scrapy for web scraping, allowing the application to stay current with the latest medical information and recommendations.

In this paper, we will discuss the methods used in Medibot, including data collection, preparation, model selection, and evaluation. We will also cover the role of model explanation in healthcare applications and how Medibot implements both global and local explanations to enhance user trust.

2. RELATED WORK

Several AI-based diagnostic tools have been developed to assist with disease prediction and diagnosis. For example, IBM Watson Health uses AI to analyze medical data, including images and text, to support clinical decision-making. Similarly, platforms like DeepAI provide AI-enabled tools for disease identification, but they often rely on single models, which limits flexibility.

A significant challenge in AI-based medical systems is the complex nature of many models, making it hard for users to understand how predictions are made. This lack of clarity can impede the adoption of AI technologies in healthcare. Local Interpretable Model-agnostic Explanations (LIME) and Shapley Additive explanations (SHAP) have become popular methods for explaining AI predictions, helping users grasp the decisions of complex models. These methods concentrate on both local and global explanations; local explanations make choices for particular predictions more understandable, while global explanations draw attention to trends throughout the entire model.

Medibot builds on existing solutions by incorporating these explanation techniques and allowing users to choose from various machine learning models. The addition of global explanations (through summary plots) and local explanations (via force plots) gives users more transparency and insight into the model's reasoning, boosting their understanding and trust in the predictions.

3. METHODOLOGY

3.1 Data Collection

The dataset for this study comes from the Mendeley repository, containing over 246,000 rows of medical data, including 773 unique diseases and 377 symptoms. This dataset is ideal for disease prediction since it includes extensive information on relationships between symptoms and diseases. Each row signifies a specific patient with a set of symptoms, enabling the model to learn the patterns of how various symptoms relate to different diseases.

The dataset is highly imbalanced, with some diseases occurring far more frequently than others. To address this issue, a preprocessing step was introduced where rare disease classes with very few instances were identified and removed from the dataset. This helped to focus the model on more prevalent diseases, ensuring better performance and reducing the impact of class imbalance during training. While this approach may limit the model's ability to predict rare diseases, it helped improve the overall efficiency and accuracy for the more common diseases in the dataset.

3.2 Data Preprocessing

Data preprocessing is a vital step in preparing the dataset for machine learning models. Key preprocessing steps include:

Handling Missing Data: Missing values were filled in with median values for numerical features and mode imputation for categorical features to ensure no valuable data points were lost.

Encoding Categorical Variables: Categorical data, such as disease names and symptoms, was encoded using one-hot encoding for nominal variables and label encoding for ordinal features.

Normalization: Numerical features were normalized to ensure all features had the same scale, which is crucial for models like K-Nearest Neighbors and Logistic Regression that are sensitive to input data scale.
Data Splitting: The dataset was divided into a training set (80%) and a testing set (20%) using stratified sampling to maintain the distribution of diseases in both sets

3.3 Model Selection

The following machine learning models were assessed for disease prediction:
Logistic Regression: A simple model often used for binary classification that was chosen for its speed and straightforwardness.
Naive Bayes: A probabilistic classifier based on Bayes' Theorem that assumes independence between features, making it efficient in terms of computation.
K-Nearest Neighbors (KNN): A non-parametric model that classifies cases based on the majority vote of adjacent instances in the feature space.
Random Forest: An ensemble method that generates multiple decision trees and combines their results to enhance accuracy.
XGBoost: A gradient-boosted tree model recognized for its high performance and scalability, especially with large datasets.
Each model was trained and tested on the preprocessed dataset. Cross-validation was utilized to evaluate model performance and minimize overfitting.

3.4 System Architecture

The system architecture of Medibot consists of three primary layers: the **Frontend Interface**, the **Backend API Layer**, and the **Machine Learning Engine**.

- **Frontend (React):** Users interact through a web-based interface built with React. They input symptoms, choose machine learning models, and view prediction results with explanations.
- **Backend (FastAPI):** The backend processes requests from the frontend. It handles input data and communicates with the machine learning models. It also updates web scraping through Scrapy.
- **Machine Learning Module:** This layer includes trained models such as Logistic Regression, Naive Bayes, KNN, Random Forest, and XGBoost. It manages prediction logic and has components for preprocessing, model inference, and providing explanations with SHAP and LIME.
- **Database and Dataset Layer:** The dataset from Mendeley is stored and managed in this layer. It regularly fetches and processes updated data.
- **Explain ability Engine:** This engine offers both global and local explanations through SHAP plots, which are integrated into the web interface for transparency.

The figure below illustrates the system architecture of Medibot:

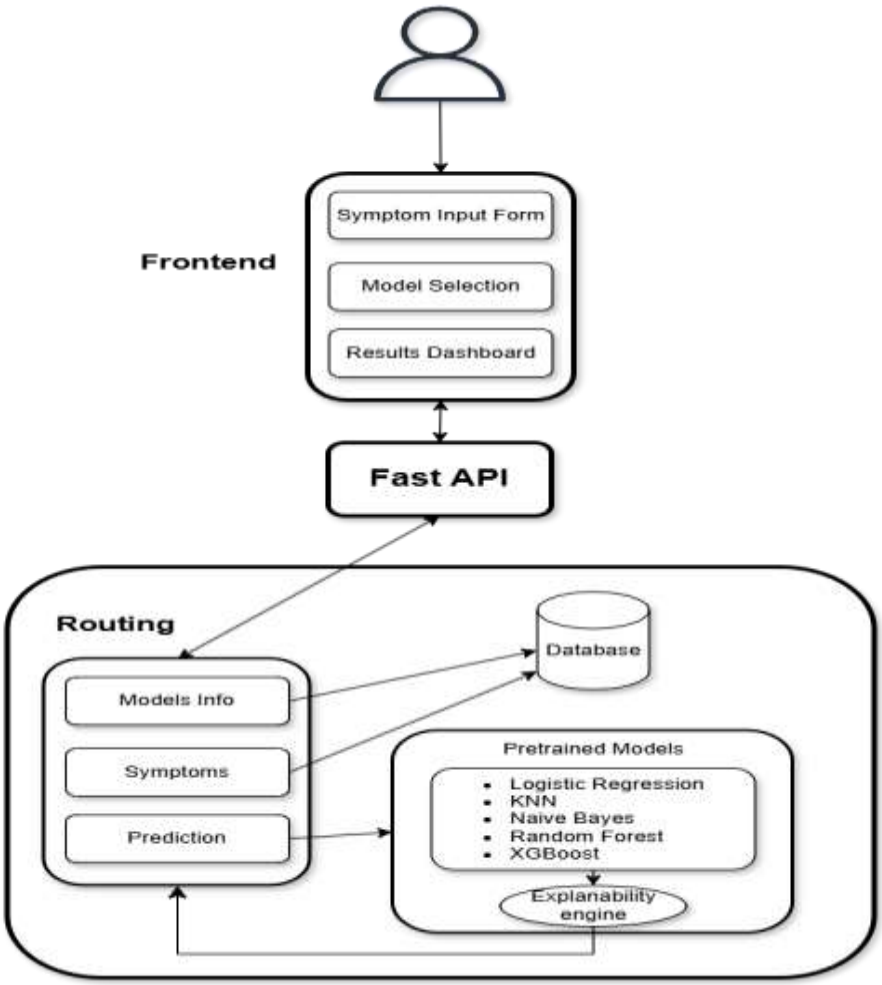

Figure 1: system architecture

4. MODEL EXPLAINABILITY

To enhance transparency and user trust, Medibot includes explainability features for its AI models.

4.1 Global Explanation (Summary Plot)

The summary plot shows the feature importances across the entire dataset, highlighting the most significant symptoms for disease prediction. This gives users insights into which symptoms are most predictive of disease outcomes.

4.2 Local Explanation (Force Plot)

For individual predictions, Medibot provides a force plot to illustrate how each symptom contributes to the model's decision. This assists users in understanding how their own symptoms affect the expected disease and helps bolster trust in the accuracy of the model's outcomes.

$$\phi_i = \sum_{S \subseteq N \setminus \{i\}} \frac{|S|!(|N| - |S| - 1)!}{|N|!} [f(S \cup \{i\}) - f(S)]$$

where ϕi is the SHAP value for feature i, N is the set of all features, and f(S) represents the model prediction when only subset S of features is present.

5. MODEL EVALUATION

Model performance was measured using these metrics:

- Accuracy: The ratio of correct predictions to all predictions made.
- Precision: The ratio of accurately anticipated positive observations to all predicted positive observations is known as precision.
- Recall: The ratio of true positive predictions to all actual positives.
- F1-Score: The harmonic mean of precision and recall, balancing the two metrics.

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

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

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

$$F_1 = 2 \times \frac{\text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}$$

Where:

- TP = True Positives
- TN = True Negatives
- FP = False Positives
- FN = False Negatives

Logistic regression formula

$$P(y = 1|x) = \frac{1}{1 + e^{-(\beta_0 + \beta_1 x_1 + \beta_2 x_2 + \cdots + \beta_n x_n)}}$$

Where, P(y=1|x) is the likelihood of a disease based on the symptoms x, while βi are the model coefficients acquired during the training process.

6. RESULTS AND DISCUSSION

The performance of Medibot showcases the potential of AI in healthcare. Logistic Regression and Naive Bayes deliver quick, accurate predictions, while more complex models like XGBoost provide higher accuracy at the expense of longer training times. The incorporation of model explainability through summary and force plots enables users to grasp the reasoning behind the predictions, addressing a major obstacle to AI adoption in medical settings.

The system's capacity to manage large datasets, provide real-time predictions, and explain its decision-making processes makes Medibot a valuable tool for early disease detection and medical decision support.

Screen shots:

Figure 2: Home page

Figure 3: User enters their symptoms

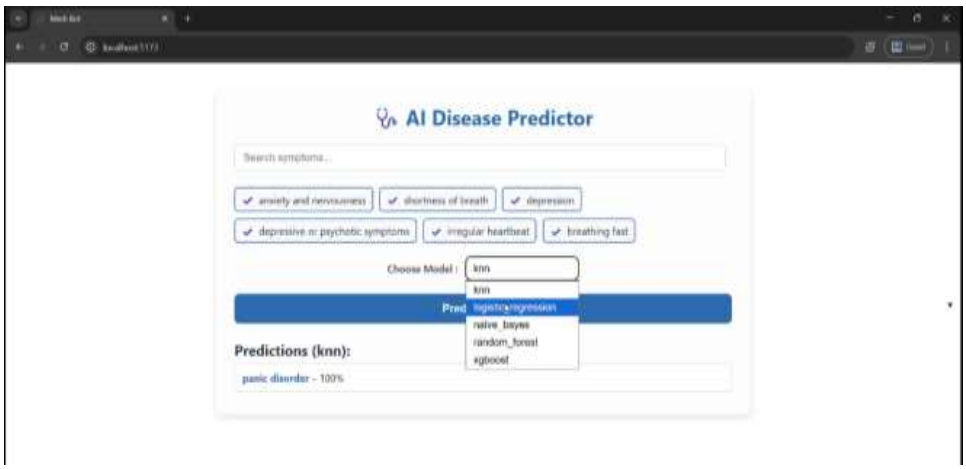

Figure 4: User selects the model to predict disease

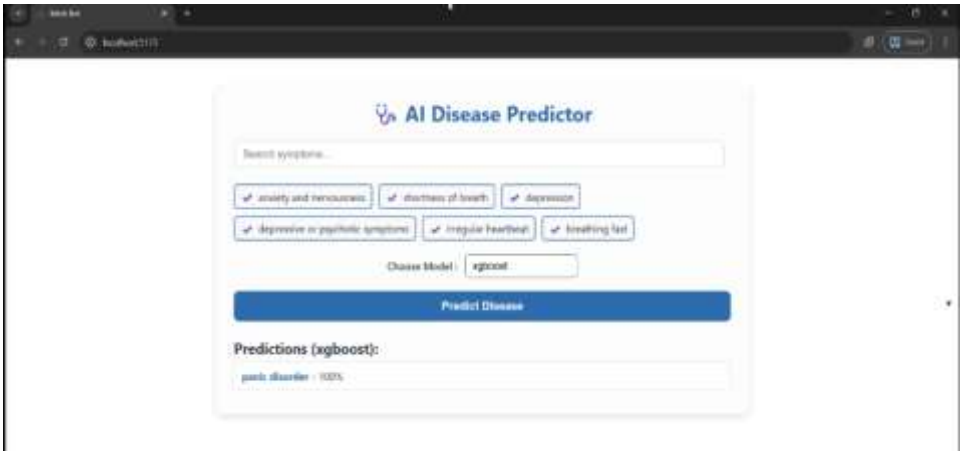

Figure 5: Based on model selection results shown below

Finally, the output for our symptoms and the predicted disease along with its accuracy is shown below:

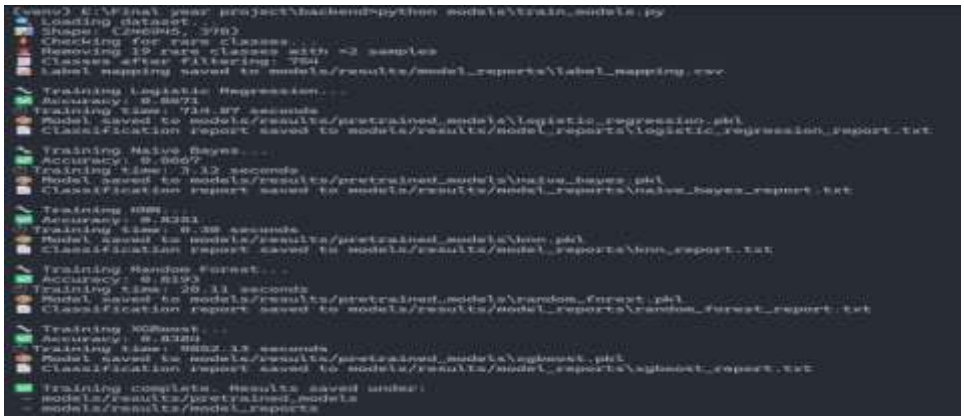

Figure 6: Accuracy of different algorithm

Model	Accuracy	Precision	Recall	F1-Score
Logistic Regression	86.71	0.8747	0.867	0.8669
Naive Bayes	86.67	0.8891	0.866	0.8697
KNN	82.51	0.8329	0.825	0.8259
Random Forest	81.93	0.8305	0.819	0.8205
XGBoost	83.84	0.8416	0.838	0.8387

Table 1: Model Performance Evaluation

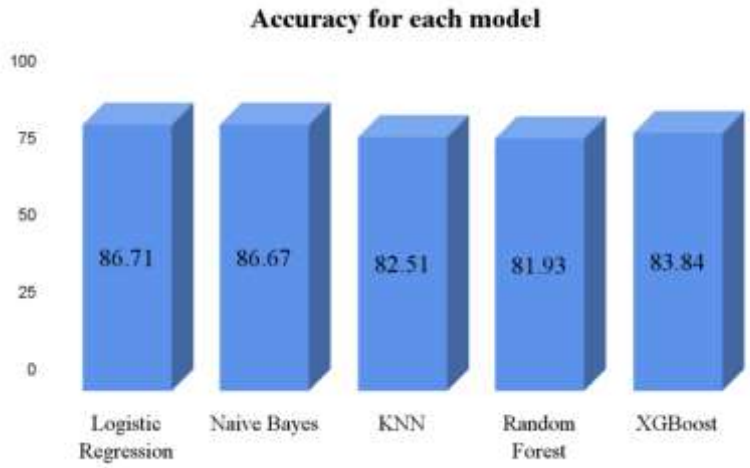

Figure 7: Comparison graph

As shown in Table 1, Naive Bayes and Logistic Regression achieved high accuracy and F1-Score. While XGBoost had a better F1-Score, its training time was significantly longer, which might limit its use in real-time scenarios.

Logistic Regression achieved the highest performance because it is inherently designed for binary input features. As the dataset represents symptoms as binary values (present or absent), the model aligns naturally with the data structure, resulting in improved prediction accuracy.

7. CONCLUSION

Medibot illustrates the impact of AI in healthcare by offering a multi-model disease prediction system with strong explainability features. With its user-friendly design and focus on transparency, Medibot allows users to make informed decisions based on AI-powered predictions. Future efforts will concentrate on expanding the dataset, enhancing model accuracy, and optimizing the system for faster real-time performance.

REFERENCES

1. D. Shrivas, K. Asati, V. Chitade, and A. Bhagwat, "Multiple disease prediction application using ML," *The International Journal of Engineering Research (TIJER)*, vol. 24, no. 11, pp. 31–45, Nov. 2023. [Online]. Available: https://tijer.org/tijer/papers/TIJER2411031.pdf
2. S. Singh, S. Agarwal, V. Singh, B. Hazela, V. Dubey, and P. Singh, "Comparative analysis of ML algorithms for multiple disease prediction," *ResearchGate*, 2023. [Online]. Available: https://www.researchgate.net/publication/394478915_Comparative_Analysis_of_Machine_Learning_Algorithms_for_Multiple_Disease_Prediction_Model_with_An_Optimized_Scalable_Deployment
3. F. Din, A. Ul Haq, M. Yaseen, A. Khan, and A. Ali, "Multi-disease prediction using ML and deep learning models," *International Journal of Healthcare Research*, vol. 32, no. 4, pp. 567–578, Dec. 2022.
4. S. S. Al-qar and A. Algar, "Disease prediction from symptom descriptions using deep learning and NLP," *ResearchGate*, 2023. [Online]. Available: https://www.researchgate.net/publication/387968668_MULTI-

DISEASE_PREDICTION_USING_MACHINE_LEARNING_AND_DEEP_LEARNING_MODELS

5. Y. You and X. Gui, "Self-diagnosis through AI-enabled chatbot-based symptom checkers," *Journal of Medical Internet Research*, 2022. [Online]. Available: https://pmc.ncbi.nlm.nih.gov/articles/PMC8075525/
6. J. Liu, H. Chen, and F. Wu, "A deep learning-based method for multi-disease classification," *Journal of Healthcare*, vol. 8, no. 4, pp. 567–578, Dec. 2020.
7. K. Patel, A. Shah, and D. Desai, "A robust system for multi-disease prediction using machine learning," *Springer Nature Computer Science*, vol. 3, no. 2, pp. 144–150, Mar. 2021.
8. F. Khan and I. Alam, "Comparative study of machine learning methods for predicting multiple diseases from medical records," *Journal of Big Data Analytics in Healthcare*, vol. 5, no. 3, pp. 123–134, Dec. 2021.
9. R. Ribeiro *et al.*, "A survey on explainable AI techniques for diagnosis and prognosis in healthcare," 2020.
10. H. Al-Mallah *et al.*, "Multiple disease prediction using hybrid deep learning architecture," in *Proc. IEEE Int. Conf. Healthcare Informatics*, 2016.
11. E. Char *et al.*, "Ethical considerations in AI-driven healthcare," 2020.
12. M. Wild *et al.*, "Streamlit for machine learning: Creating interactive web apps in Python," *Journal of Machine Learning Tools*, vol. 12, no. 1, pp. 23–35, 2022.
13. C. Chauhan *et al.*, "Multiple disease prediction using machine learning algorithms," *International Journal of Computational Health Sciences*, vol. 10, no. 2, pp. 98–110, 2021.
14. A. Kamboj *et al.*, "A machine learning model for early prediction of multiple diseases to cure lives," *AI Healthcare Journal*, vol. 6, no. 1, pp. 45–55, 2020.
15. A. K. Gupta and M. Mehra, "A review on predictive models for multiple diseases utilizing data mining techniques," *International Journal of Data Mining and Bioinformatics*, vol. 14, no. 1, pp. 123–145, Feb. 2020.
16. P. K. Kumar, A. Dutta, and P. Kumar, "Application of graph mining algorithms for the analysis of web data," *SSRN Electronic Journal*, 2023. doi: https://doi.org/10.2139/ssrn.4365862.