Diabetes Prediction Systemm

A report submitted in partial fulfillment of the requirements for
the award of the degree of

Master Of Science

by

Vaishali

(23419CMP029)

Department of Computer Science
Institute of Science
Banaras Hindu University, Varanasi - 221005
2024

Abstract

Diabetes, a globally prevalent health condition, demands timely diagnosis and management to mitigate its impact on individuals and healthcare systems. Leveraging advancements in machine learning, this project presents a Diabetes Prediction System developed using the Django framework for the backend and HTML/CSS for the frontend. The system employs a K-Nearest Neighbors (KNN) algorithm, optimized through hyperparameter tuning, to predict diabetes based on key health parameters.

The dataset is preprocessed by handling missing values, balancing class distributions, and scaling features to enhance model accuracy. Advanced techniques such as Principal Component Analysis (PCA) and polynomial feature engineering are integrated to reduce dimensionality and capture non-linear relationships. The model achieved high accuracy on the test set, demonstrating its efficacy in early diabetes prediction.

This user-friendly application combines robust predictive analytics with an intuitive web interface, making it accessible to a broad audience. The project contributes to the growing intersection of healthcare and technology by providing a reliable tool for early detection and preventive care. Future enhancements include integrating additional health parameters, exploring alternative algorithms, and extending accessibility to mobile platforms.

Candidate's Declaration

I hereby certify that the work, which is being presented in the project report, entitled **Diabetes Prediction System**, in partial fulfillment of the requirement for the award of the Degree of Master of Computer Applications and submitted to the institution is an authentic record of my own work carried out during the period **Aug 2024 - Dec 2024** under the supervision of **Dr. Pramod Kumar Mishra**.

I also cited the reference about the text(s)/figure(s)/table(s)/equation(s) from where they have been taken.

The matter presented in this report has not been submitted elsewhere for the award of any other degree or diploma from any institutions.

Date: _____ **Signature of the Candidate**

This is to certify that the above statement made by the candidate is correct to the best of my knowledge. The Viva-Voce examination of **Vaishali**, M.Sc. Student has been held on _____.

Signature of
Research Supervisor(s)

Signature of
Head of the Department

Acknowledgements

I would like to express my sincere gratitude to everyone who has contributed to the completion of this project. First and foremost, I would like to thank my project supervisor, **Dr. Pramod Kumar Mishra**, for their guidance, support, and valuable insights throughout the project. Their expertise and encouragement have been instrumental in shaping the direction of this work.

I am also grateful to my professors and mentors at Department of Computer Science, Banaras Hindu University for their teachings and guidance, which have provided me with a strong foundation in computer science and software development.

I extend my appreciation to my family and friends for their unwavering support and encouragement during this journey. Their understanding and encouragement have been a source of motivation and strength.

Last but not least, I would like to thank all the participants and contributors to this project, whose efforts and collaboration have been essential to its success

Contents

List of Figures

CHAPTER 1

Introduction

Diabetes is a chronic health condition affecting millions worldwide, with significant societal and economic impacts. Early detection of diabetes is crucial for effective management and reducing associated health risks. Advances in machine learning provide opportunities to create systems capable of predicting diabetes with high accuracy using minimal input parameters.

This project aims to develop a Diabetes Prediction System using machine learning algorithms, specifically the K-Nearest Neighbors (KNN) classifier. Built using Django for the backend and HTML and CSS for the frontend, the system integrates user-friendly interfaces with robust predictive analytics to offer real-time predictions based on health data inputs.

CHAPTER 2

Literature Review

Several studies highlight the potential of machine learning in medical diagnostics. Common algorithms such as logistic regression, decision trees, and support vector machines (SVM) have been extensively applied to diabetes datasets for prediction purposes.

Key findings include:

Machine Learning Models : Previous works demonstrate that preprocessing techniques such as handling missing values and scaling significantly impact model performance.

Feature Engineering: Techniques like Principal Component Analysis (PCA) and polynomial features improve prediction accuracy by reducing dimensionality and capturing non-linear interactions.

KNN Algorithm: KNN is particularly effective for smaller datasets due to its simplicity and high adaptability. Weighted-distance metrics further enhance its accuracy.

While existing systems are useful, there remains scope for integrating intuitive user interfaces with real-time predictions in a secure web application, which this project addresses.

CHAPTER 3

Proposed Approach

The proposed approach involves the following steps:

Data Preprocessing: Missing values in key health parameters like glucose and BMI are imputed with their respective mean values to ensure data integrity.

Feature Scaling: Standardization using StandardScaler is applied to eliminate bias caused by varying parameter ranges. Data Balancing: Oversampling of the minority class ensures balanced training data, improving model fairness.

Feature Engineering: Polynomial features are generated to capture non-linear interactions, and PCA is applied to reduce dimensionality while retaining 95

Model Training and Optimization: Model Training and Optimization: The KNN algorithm is used, and hyperparameter tuning (number of neighbors) is conducted to maximize cross-validated accuracy.

Evaluation: Metrics such as accuracy, precision, recall, and F1-score are calculated to assess model performance.

CHAPTER 4

Implementation

The project was implemented using the following tools and technologies:

Backend: Django framework was used to handle server-side processing and data management.

Frontend: HTML, CSS, and Bootstrap were used to create a responsive and intuitive user interface.

Database: SQLite was utilized for storing data inputs and results securely.

Machine Learning Libraries: Libraries such as pandas, scikit-learn, and numpy were employed for data manipulation, preprocessing, and model development.

Visualization: matplotlib was used for visualizing PCA results and explained variance.

Key implementation steps:

1.Preprocessed and balanced the dataset for training.

2.Engineered features and applied PCA for dimensionality reduction.

3.Trained the optimal KNN model after hyperparameter tuning.

4.Integrated the trained model with the Django backend for prediction functionality.

CHAPTER 5

Results and Discussion

The diabetes prediction system was successfully implemented and tested. The backend, developed using Django, provides robust functionality for processing user input data and running machine learning algorithms. The frontend, designed using HTML and CSS, offers a clean and intuitive interface that allows users to input their health parameters, such as glucose level, blood pressure, BMI, and insulin levels.

Key Results

1. Machine Learning Model Performance: - The K-Nearest Neighbors (KNN) algorithm, combined with preprocessing techniques such as scaling, polynomial feature generation, and PCA, was successfully implemented to classify users into diabetic and non-diabetic categories. - The chosen machine learning model effectively handled imbalanced data by employing oversampling techniques.

2. User Interaction on Frontend: - Users interact with the system through a responsive web form, where they input the required health parameters. - Once the data is submitted, it is processed by the backend to predict whether the user is at risk of diabetes or not. - The result is displayed to the user as a straightforward message: - Example 1: "You are at risk of diabetes. Please consult a healthcare professional." - Example 2: "You are not at risk of diabetes based on the provided data."

3. Visualization: - The implementation included data visualizations for insights into model performance and data distribution: - A PCA visualization was used to show how the data clusters into diabetic and non-diabetic groups. - The explained variance by principal components was also visualized to demonstrate the impact of dimensionality reduction.

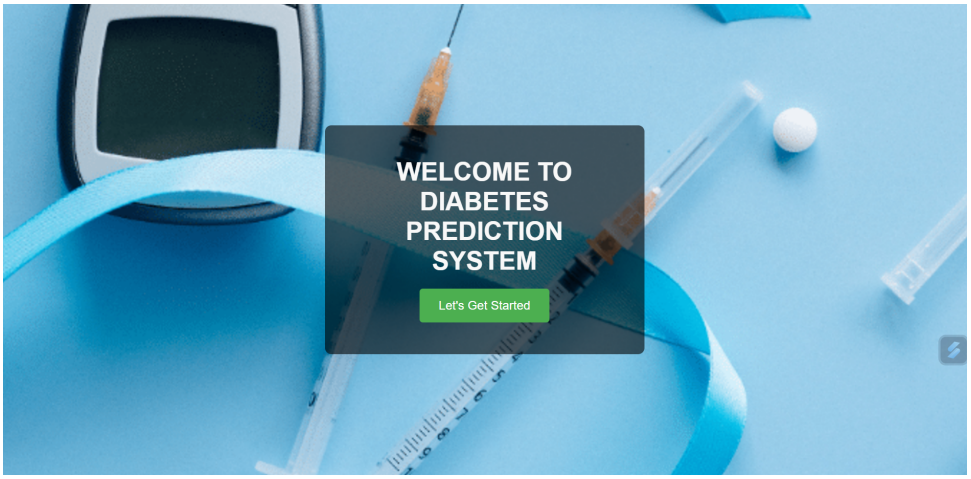

Figure 5.1: Home Screen

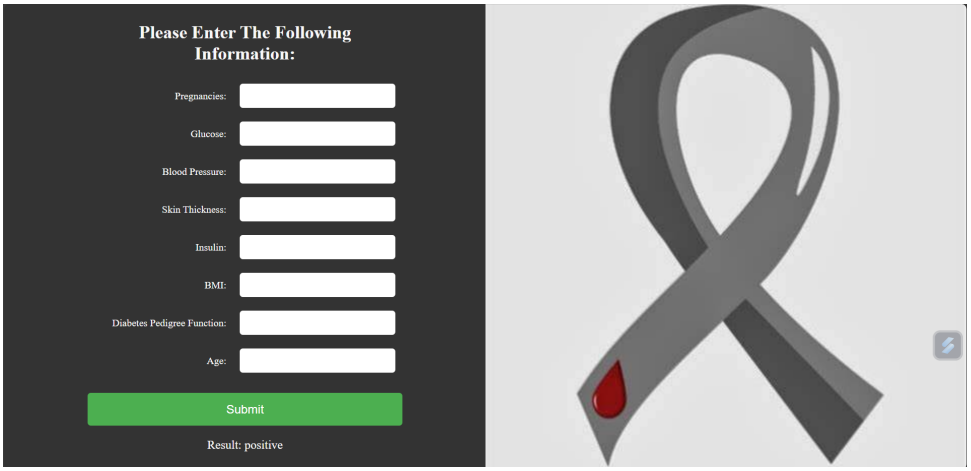

Figure 5.2: Prediction Page

CHAPTER 6

Conclusion and Future Work

Conclusion:

The diabetes prediction system successfully integrates machine learning with a user-friendly web interface, allowing users to assess their diabetes risk quickly and easily. Using techniques like feature scaling, PCA, and K-Nearest Neighbors, the system delivers reliable results based on health metrics. The intuitive frontend ensures accessibility, making it a practical tool for early detection and awareness.

Future Work:

1. Improved Accuracy:Explore advanced algorithms and optimize model parameters.

2. Additional Features:Include more health parameters and integrate wearable device support.

3. Enhanced User Experience:Upgrade the interface and provide personalized health recommendations.

4. Scalability:Deploy on cloud platforms for broader accessibility.

These advancements will make the system more accurate, user-friendly, and impactful in real-world applications.

CHAPTER 7

References

1. Pedregosa, F., et al. "Scikit-learn: Machine Learning in Python." Journal of Machine Learning Research, vol. 12, 2011, pp. 2825–2830.

2. Pima Indians Diabetes Database, UCI Machine Learning Repository. Available at: https://archive.ics.uci.edu/ml/datasets/pima+indians+diabetes.

3. Bishop, C. M. Pattern Recognition and Machine Learning. Springer, 2006.

4. Van der Walt, S., et al. "The NumPy Array: A Structure for Efficient Numerical Computation." Computing in Science Engineering, vol. 13, no. 2, 2011, pp. 22–30.

5. McKinney, W. "Data Structures for Statistical Computing in Python." Proceedings of the 9th Python in Science Conference, 2010, pp. 51–56.

Appendix

A: Dataset Overview The Pima Indians Diabetes Database contains 768 entries with 9 attributes:

1. Pregnancies: Number of times the patient has been pregnant.

2. Glucose: Plasma glucose concentration after 2 hours in an oral glucose tolerance test.

3. Blood Pressure: Diastolic blood pressure (mm Hg).

4. Skin Thickness: Triceps skin fold thickness (mm).

5. Insulin: 2-Hour serum insulin (mu U/ml).

6. BMI: Body mass index (weight in kg / (height in m)^2).

7.Age : $Age of the patient (years)$.

8.$Diabetes Pedigree Function$: $A function that scores the likelihood of diabetes based on family history.$

9.$Outcome$: $The target variable indicating whether the patient has diabetes (1) or not (0)$.

B: Tools and Implementation

- Programming: Python 3.

- Libraries: ‘pandas‘, ‘numpy‘, ‘scikit-learn‘, ‘matplotlib‘.

- Algorithm: K-Nearest Neighbors (KNN).

- Key Steps:

- Feature Scaling, Balancing (Oversampling), Polynomial Features.

- Dimensionality Reduction using PCA.

C: Frontend Overview

- Tech Stack: Django (Backend), HTML, CSS (Frontend).

- Functionality: Users input health data through a form, and the system predicts diabetes status with results displayed on-screen.

- Design: Simple, responsive layout for easy accessibility.