GlycoDetect a Diabetic Prediction Model using ML

Archana Nikose[1], Harsh Kuite[2], Kalyani Mude[2], Aniruddha Polke[2], Nikita Nanhe[2]

[1]*Assistant Professor, Department of CSE, Priyadarshini Bhagwati College of Engineering*
[2]*Student, Department of CSE, Priyadarshini Bhagwati College of Engineering*

---***---

Abstract - This project focuses on using machine learning to predict a patient's risk of developing diabetes based on a test. This database, compiled by the National Institute of Diabetes and Digestive and Kidney Diseases, contains health indicators for Pima Indian patients. The project involves several key steps: initial data, selecting a feature, selecting a model, updating the hyperparameter, and deploying it via the Flask web application. In the data preprocessing phase, feature scaling and normalization are used to standardize the dataset, while missing values and outliers are handled to ensure data integrity. Feature selection uses correlation matrix and recursive feature elimination (RFE) to reduce dimensionality and improve model efficiency. To ensure the model is optimized for latent data, the dataset is split into two parts: 66% for training and 34% for testing. Various machine learning algorithms are evaluated, including logistic regression, naive Bayes, K-nearest neighbours, decision trees, and support vector classifiers. Logistic regression was selected as the final model due to its accuracy on the test data (80.53%). The model uses grid search for hyperparameter tuning to improve its performance. The training model is embedded in the Flask web application, allowing users to access health metrics and get real-time estimates of blood pressure. The system is designed to be user-friendly and scalable, providing a practical tool for early diagnosis of diabetes. All methods ensure that the model is accurate, reliable, and capable of making real-world predictions.

***Keywords*:** prediction, diabetes, glucoses, insulin, machine learning, logistic regression, naive bayes, k-nearest neighbours, decision tree, support vector classifier.

1. INTRODUCTION

In this day and age, one of the most notorious ailments to have taken the world by storm is Diabetes, which is a disease which causes an increase in blood glucose levels as a result of the nonappearance or moo levels of Attack. Since there are various procedures to analyse the calm with the ailment, the examination and figure can be bewildering and, in a few cases, questionable. It is incomprehensible to capture it at the level. Diabetes is extending day by day in the world since of characteristic, genetic components. The numbers are rising rapidly due to a few factors which joins terrible nourishments, physical idleness and various more. Diabetes is a hormonal clutter in which the disappointment of the body to provide insult causes the absorption framework of sugar in the body to be unordinary, in this way, raising the blood glucose levels in the body of a particular individual. Unequivocally starvation, thirst and visit urination are a few of the discernible characteristics. Certain chance factors such as age, BMI, Glucose Levels, Blood Weight, etc., play a basic portion to the commitment of the disease. Diabetes is situated as the fifth deadliest illness worldwide.

According to the WHO report India is situated No. 1 with 31.7 million no. of diabetic calm in 2000 and is likely to increase up to 79.4 million. Since there is a huge dangerous likelihood of extending no. of diabetic diligent where exact assurance will be the require of the hour. Along these lines this estimation gives the motivation to carry out the explore to find the best performing calculation. In this paper we will make utilize of four coordinated learning calculation and perform comparison on their result to propose the best approach to expect the diabetes which will be strong for making taught choices accurately.

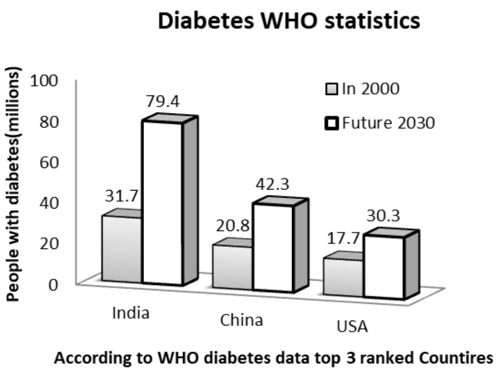


Figure 1.1: WHO Diabetes statistics

Additionally, a consider by the World Prosperity Organization appears that in 2012, the sickness particularly caused the passing of 1.6 million people. Right presently, the helpful condition has no enduring cure. In any case, the reasonability of managing it is subordinate on its early assurance. Concurring to the U.S Division of Prosperity and Human Organizations, early conclusion of diabetes is crucial in keeping patients with the ailment sound. In show disdain toward of the fundament of early conclusion and the breakthrough in its organization, the right presently open gadgets are unable for advantageous assurance. The quick enhancement of computational bits of knowledge has made it conceivable to increase the conclusion precision and to help the plan. Computational experiences methods are utilized to consider plans and conduct of the remedial condition and to build a correct basis that is fitting in diagnosing the sickness. The exactness and capability of diabetes assurance utilizing computational experiences strategies move with the sort of classification calculation utilized, and the sum, quality, and accuracy of the planning dataset.

Unfortunately, this disease cannot be slaughtered. But we can control it by restricting the glucose level in blood. When diabetes is distinguished, its affect can be minimized. But this is not a basic task. To recognize the disease, data are taken from patients like insult, age, body mass list, family history of diabetes, etc. and at that point counselled to a pro. At that point the master takes a choice utilizing his/her data and experience. But this recognizing confirmation handle is outstandingly time using and in a few cases most over the top. A few of the time, it additionally cheats the conclusion handle due to the require of inclusion of the pros. Computer mechanized assurance can play a basic character in the disclosure of diabetic patients.

2. LITERATURE SURVEY

An Effective Diabetes Prediction System Using Machine Learning Techniques (2020): The research conducted by S. M. Mahedy Hasan, M. F. Rabbi, A. I. Champa, and M. A. Zaman focuses on designing a diabetes prediction system by employing a Tree-Based machine learning model. The primary algorithm used in this study is the Extra Tree algorithm, known for its ability to handle large datasets efficiently by constructing multiple decision trees. To further enhance the system's predictive accuracy, the researchers applied the AdaBoost classifier, a boosting technique that combines weak classifiers to create a strong one. The combination of these methods is effective in identifying diabetes with improved precision. The system leverages advanced machine learning approaches, demonstrating that it can significantly improve the early detection of diabetes, ultimately aiding healthcare professionals in timely intervention.[1]

Prediction of Diabetes Using Machine Learning Algorithms in Healthcare (2022): In this study, M. A. Sarwar, W. N. Kamal, M. Hamid, and M. A. Shah explore various machine learning algorithms to predict diabetes from patient medical records. The research focuses on comparing algorithms to find the most accurate for early diagnosis. The algorithms considered include support vector machines, decision trees, random forests, and others. By analyzing the strengths and weaknesses of each algorithm, the study identifies the best-performing model for diabetes prediction. The paper highlights the growing importance of data-driven approaches in the healthcare sector, demonstrating that machine learning algorithms can significantly improve the diagnostic process. The study also emphasizes the potential of integrating machine learning with patient records to enhance the accuracy of diabetes diagnosis, thus reducing the risks associated with delayed detection.[2]

Diabetes Prediction using ML with Feature Selection and Dimensionality Reduction (2022): Sivaranjani S., Ananya S., Aravinth J., and Karthika R. propose a machine learning approach to diabetes prediction, focusing on feature selection and dimensionality reduction techniques. Feature selection helps identify the most relevant features from a dataset, eliminating unnecessary or redundant information that could slow down or reduce the accuracy of the model. Dimensionality reduction further simplifies the data by projecting it into a lower-dimensional space, making it easier to process without losing critical information. The combination of these techniques improves the performance of machine learning models, resulting in faster and more accurate predictions. This study highlights the significance of preprocessing techniques like feature selection and dimensionality reduction in improving machine learning models for healthcare applications. By optimizing the input data, these methods enhance the overall efficiency and accuracy of diabetes prediction systems.[3]

Diabetes Prediction Model Using Data Mining Techniques (2021): R. Rastogi and M. Bansal developed a diabetes prediction model by leveraging data mining techniques. Their research utilizes a dataset from Kaggle, a popular platform for data science competitions, to build a prediction model with Python. The study applied various data mining and machine learning algorithms to analyze the dataset and develop a robust prediction model. The model's performance was validated through extensive testing using performance metrics such as accuracy, precision, recall, and F1-score. The study confirmed the model’s effectiveness for early diagnosis of diabetes, proving that the application of data mining techniques could enhance predictive analytics in healthcare. The paper highlights the potential of using publicly available datasets and open-source tools like Python to develop scalable and efficient healthcare solutions. Furthermore, it demonstrates the critical role that data mining can play in processing large volumes of health data and identifying patterns that may not be immediately apparent to clinicians.[4]

3. PROPOSED METHODOLOGY

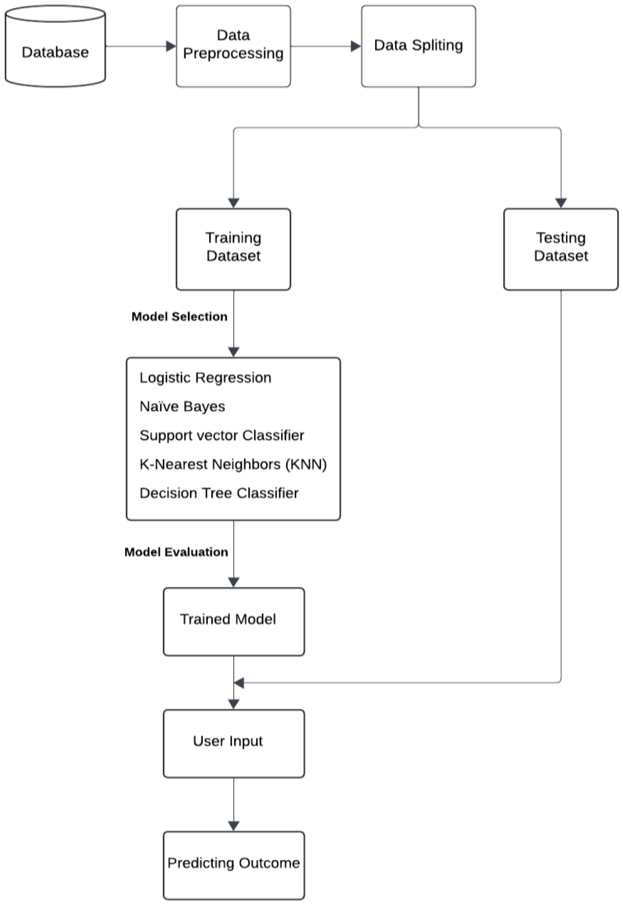
Figure 3.1: ML Model Development Flowchart

The flowchart represents a complete machine learning pipeline, starting from data extraction and ending with predictions based on user input. Initially, raw data is collected from a database and then undergoes preprocessing, which addresses issues like missing values and inconsistencies. This step ensures that the data is clean and suitable for model training. Once processed, the data is split into two subsets: a training set and a testing set. The training data is used to build machine learning models, while the testing data is reserved for evaluating the model’s performance, ensuring that it can generalize well to unseen data.

Several machine learning algorithms are tested on the training set, including Logistic Regression, Naive Bayes, Support Vector Classifier (SVC), K-Nearest Neighbours (KNN), and Decision Tree Classifier. Each algorithm is evaluated based on performance metrics like accuracy and precision, allowing the selection of the best-performing model. Once the model is trained and evaluated, it is finalized and used to make predictions. User input is then pre-processed in the same manner as the training data, fed into the trained model, and used to predict outcomes, such as disease diagnosis or other classifications. This pipeline ensures a structured approach to developing accurate and reliable predictive models.

4. MODULES

4.1. Dataset Used



Glucose	BP	Skin Thickness	Insulin	BMI	DPF	Age
148	72	35	0	33.6	0.627	50
85	66	29	0	26.6	0.351	31
183	64	0	0	23.3	0.672	32
89	66	23	94	28.1	0.167	21
137	40	35	168	43.1	0.471	33

Table 4.1: Sample Dataset

The dataset sample provided represents patient health measurements commonly used for predicting diabetes risk. It includes seven key features: Glucose level, Blood Pressure (BP), Skin Thickness, Insulin levels, Body Mass Index (BMI), Diabetes Pedigree Function (DPF), and Age. Each row represents a patient's medical data, where values like glucose and insulin levels indicate potential risk factors for diabetes.

4.2. Data Preprocessing

Data preprocessing is the foundational and essential step in the machine learning pipeline. It emphasizes cleaning, transforming, and preparing data to ensure it's ready for effective model training. Since raw data is often incomplete or noisy, it cannot be directly used by machine learning algorithms. This phase involves detailed cleaning and preparation to guarantee high data quality and consistency. The dataset in this project comes from the National Institute of Diabetes and Digestive and Kidney Diseases and includes the following features:

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

Blood Pressure: Diastolic blood pressure measured in mm Hg.

Skin Thickness: Measurement of the triceps skin fold thickness, expressed in millimeters (mm).

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

BMI: Body mass index, determined by dividing a person's weight (in kilograms) by the square of their height (in meters).

Diabetes Pedigree Function: A metric that assesses the likelihood of diabetes based on family history.

Age: The patient's age.

Outcome: A binary variable indicating diabetes diagnosis (1 for positive, 0 for negative).

4.2.1. Handling Missing Values and Outliers

Imputing missing data using appropriate strategies is essential to avoid gaps and ensure data consistency. Addressing missing values and detecting outliers are key steps to guarantee that the model is trained on reliable and representative data:

Missing Data: Any missing values in the dataset are dealt with by imputing them using statistical methods such as the mean or median. For instance, missing values in features like glucose or insulin are replaced with the median of the respective column to preserve data integrity.

Outlier Management: Outliers can skew the performance of machine learning models. The Interquartile Range (IQR) technique is employed to detect and handle outliers in crucial features such as glucose and BMI. These outliers are either removed or adjusted to reduce their negative impact on model accuracy.

4.2.2. Feature Encoding

As all the features in this dataset are numerical, there is no need for categorical encoding. However, in many real-world applications, techniques like label encoding or target encoding are applied to convert categorical variables into numerical values.

4.2.3. Feature Selection

Feature selection is crucial for enhancing model efficiency and reducing complexity:

Correlation Matrix: A correlation matrix is computed to evaluate the relationships between various features. Highly correlated features are identified and either removed or merged to prevent multicollinearity, which can introduce instability into the model.

Heatmap: A heatmap is generated based on the correlation matrix to visualize feature relationships. This aids in identifying redundant features that can be removed without compromising the model's accuracy.

Recursive Feature Elimination (RFE): RFE is employed to iteratively eliminate less important features, reducing the dataset's dimensionality and helping the model generalize more effectively.

4.2.4. Feature Scaling and Transformation

Feature scaling is essential for improving the performance of machine learning models:

Standard Scaler: The features are standardized using scikit-learns Standard Scaler to ensure each feature has a mean of 0 and a standard deviation of 1. This step is especially important for algorithms like Logistic Regression, which are sensitive to feature scaling.

Normalization: Normalization rescales the features to a specific range (e.g., 0 to 1), preventing larger features from overpowering smaller ones and ensuring balanced model performance.

4.3. Model Selection and Training

After preprocessing the dataset, various machine learning models are trained to find the best-performing algorithm for diabetes prediction.

4.3.1. Data Splitting

Train-Test Split:

The dataset is split into two parts to evaluate the performance of the models effectively:

- **Training Data (66%)**: This portion of the dataset is used to train the machine learning model. The model learns the relationships between the features and the outcome based on this data.
- **Testing Data (34%)**: This part is reserved to evaluate the model's performance on unseen data, simulating how the model would behave with new, real-world data.

A typical split of 70% for training and 30% for testing is often used, but in this case, a 66%-34% split has been chosen to match the dataset's characteristics and model complexity.

4.3.2. Model Selection

A wide range of machine learning algorithms are available, each with its own strengths and weaknesses. This step involves carefully considering the nature of the problem and the characteristics of the data to select the most suitable algorithms. Several machine learning models are considered for this project, including:

Logistic Regression: A widely-used statistical method for binary classification tasks. It offers good interpretability and performs well with the scaled numerical features in the dataset.

Naive Bayes Classifier: A probabilistic model based on Bayes' theorem. It assumes that the features are independent, which may not always be the case in real-world datasets but can perform well with simple datasets.

K-Nearest Neighbors (KNN): A distance-based classification algorithm that assigns a class label based on the majority class of the nearest neighbors in the feature space.

Decision Tree Classifier: A model that uses a tree structure to recursively split the data based on feature values. It is easy to interpret and handles both numerical and categorical variables effectively.

Support Vector Classifier (SVC): This algorithm finds the optimal hyperplane that separates different classes. It can handle complex, non-linear relationships by utilizing kernel functions.

4.3.3 Trained Model

The model that exhibits the highest performance on the testing data is chosen as the final model. This model is then prepared for deployment, ready to make predictions on new, unseen data.

4.3.4. Model Evaluation

Each of the selected models is evaluated using various metrics to determine the best performer:

Accuracy: Indicates the ratio of correct predictions to the total number of predictions made by the model.

Confusion Matrix: Provides insight into the model's precision, recall, and F1-score by illustrating the types of errors it makes and giving a clearer picture of its overall performance.

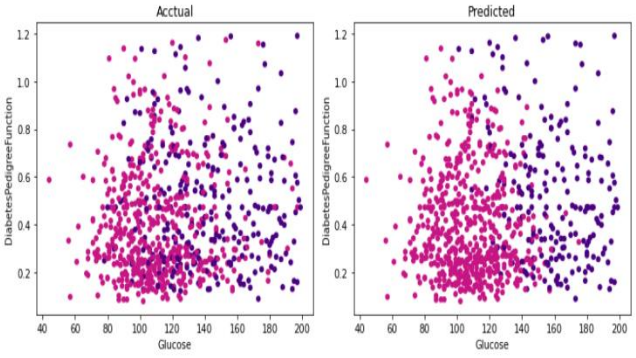


Figure 4.1: Visualizing Diabetes Prediction Accuracy

The accuracy of the models is as follows:

Algorithms	Accuracy Values
Naïve Bayes	0.7709
Support vector Classifier	0.7977
K-Nearest Neighbors (KNN)	0.7366
Logistic Regression	0.8053
Decision Tree Classifier	0.7404

Table 4.2: Accuracy of the Models

Given that Logistic Regression performs the best with an accuracy of 0.8053, it is selected as the final model.

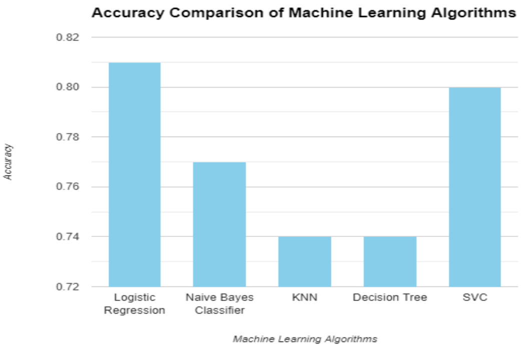


Figure 4.2: Accuracy Comparison of ML model

4.4. Hyperparameter Tuning

Hyperparameter tuning is utilized to enhance the performance of the Logistic Regression model:

Grid Search: A grid search is performed to identify the optimal combination of hyperparameters for the Logistic Regression model. This process tests various combinations of solvers (such as newton-cg, lb fgs, and lib linear), penalty terms (l2), and regularization values (C = 100, 10, 1.0, 0.1, 0.01). After tuning, the Logistic Regression model achieves a better accuracy of 0.7760 on the complete dataset, making it the final model for deployment.

4.5. Flask Application Deployment

Once the model is finalized, the next step is to deploy it as a web application using Flask, a lightweight Python framework for building web applications.

4.5.1. Web Application Structure

The Flask web application consists of the following components:

app.py: This is the main script that defines the Flask routes and handles the backend logic of the web app. The routes handle user inputs, preprocess the input data, and return predictions to the user.

HTML Templates: These templates (index.html and result.html) form the frontend of the web app. index.html contains the input form for the user, while result.html displays the prediction result.

4.5.2. User Input and Processing

Imputing missing values with appropriate techniques to prevent data gaps. When the user submits their health data through the web form (e.g., glucose levels, BMI, age), the Flask app processes this input:

Data Preprocessing: The input data is first scaled using the saved Standard Scaler object (scaler.pkl) to ensure consistency between training and prediction phases.

Prediction: The preprocessed input data is fed into the trained Logistic Regression model (lr.pkl), which outputs a binary prediction (1 for diabetes-positive, 0 for diabetes-negative).

Displaying Results: The prediction is displayed to the user via the result.html template, with a message indicating whether the user is likely to have diabetes based on the input data.

5. EXPERIMENTAL RESULT

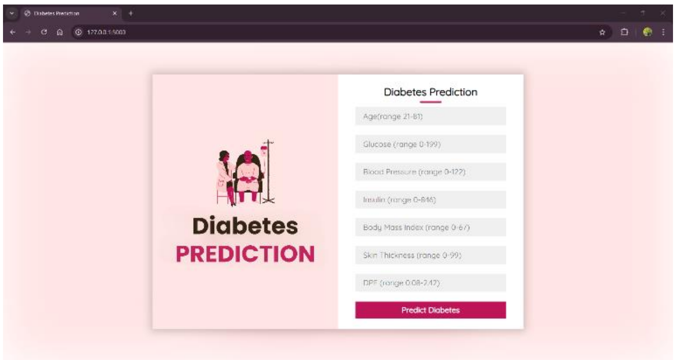
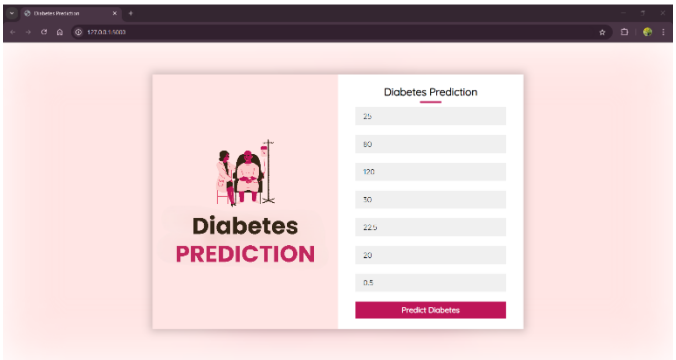


Figure 5.1: User Interface for Diabetes Prediction

This image shows the main interface of the web application, where users can input their health metrics for diabetes prediction. The homepage has a simple and user-friendly design, ensuring ease of use for people of all backgrounds. The page likely includes fields for inputting various health parameters such as glucose levels, BMI, and blood pressure, with a submit button to send the data for prediction.


Diabetes Prediction

25
80
120
30
22.5
20
0.5

Figure 5.2: Sample Input Data for Prediction

This image displays a sample of how a user would input their health data into the application. It shows fields where the user enters values like glucose level, age, and BMI. This input form collects the necessary health metrics, which are processed by the machine learning model to predict the likelihood of diabetes.

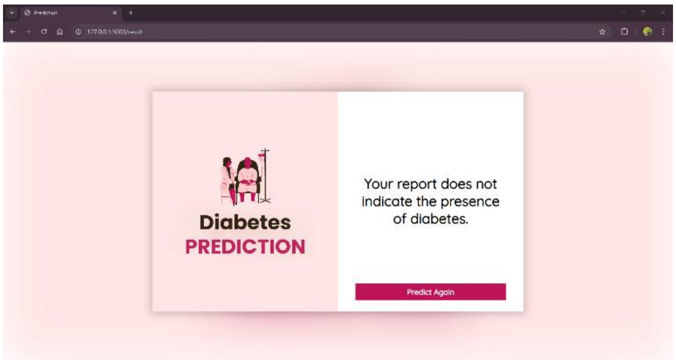
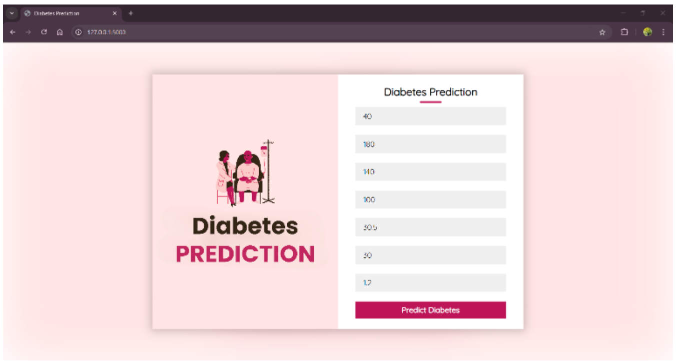

Your report does not indicate the presence of diabetes.

Figure 5.3: Diabetes Prediction Result: No Diabetes

This figure presents the result where the machine learning model predicts that the user does *not* have diabetes. After inputting the health metrics, the model processes the data and returns a prediction, which is clearly displayed to the user. The result page is designed to give a straightforward outcome, reducing any confusion about the prediction.


Diabetes Prediction

40
180
140
100
30.5
30
1.7

Figure 5.4: Sample Input Data for Prediction

Similar to Figure 5.2, this image once again illustrates a different set of sample input data for prediction. It shows how the user enters a new set of health parameters into the system, ensuring the model is tested with diverse inputs. This is important for showing how the application handles multiple user queries.

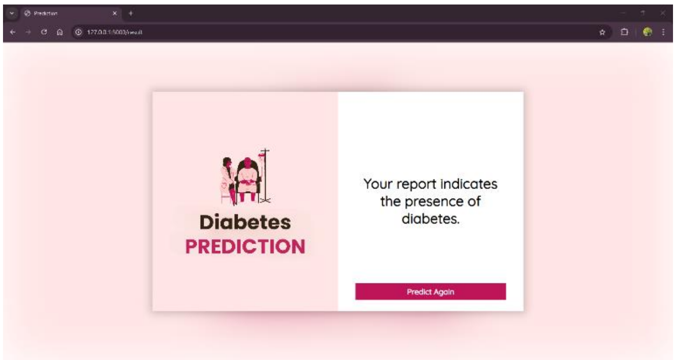

Your report indicates the presence of diabetes.

Figure 5.5: Diabetes Prediction Result: Diabetes

This image shows the prediction result when the machine learning model detects a high risk of diabetes. The prediction outcome is clearly displayed as "Diabetes," informing the user that they might be at risk. This result page is crucial as it offers guidance for users who may need to seek medical attention based on the prediction.

6. CONCLUSION AND FUTURE WORK

In conclusion, the diabetes expectation demonstrates utilizing machine learning offers a promising instrument for making strides early determination and personalized care in diabetes administration. By leveraging progressed analytics and data-driven experiences, it has the potential to change diabetes avoidance techniques, upgrade understanding results, and contribute to the broader objective of diminishing the worldwide burden of diabetes.

In this project, we proposed a web application for the fruitful prediction of Diabetes Disease. From distinctive machine learning calculations which give us most noteworthy precision on Indian Pima Dataset. As we have proposed and created an approach for diabetes malady forecast utilizing machine learning calculation, it has critical potential in the field of restorative science for the discovery of different therapeutic information precisely.

The improvement of a diabetes expectation demonstrates utilizing machine learning speaks to a noteworthy progression in the field of healthcare analytics. This demonstrate has the potential to upgrade early location and anticipation techniques for diabetes, eventually moving forward quiet results and lessening healthcare costs.

In the future, more accurate algorithms can be explored to enhance the model's performance. Additionally, incorporating more parameters such as symptoms and family history could improve the predictive power of the system, allowing for a more comprehensive diabetes risk assessment.

REFERENCES

[1] Mahedy Hasan, S. M., Rabbi, M. F., Champa, A. I., & Zaman, M. A. (2020). An Effective Diabetes Prediction System Using Machine Learning Techniques. 2020 2nd International Conference on Advanced Information and Communication Technology (ICAICT). doi:10.1109/icaict51780.2020.9333497

[2] M. A. Sarwar, N. Kamal, W. Hamid and M. A. Shah, "Prediction of Diabetes Using Machine Learning Algorithms in Healthcare," 2018 24th International Conference on Automation and Computing (ICAC), Newcastle Upon Tyne, UK, 2018, pp. 1-6, doi: 10.23919/IConAC.2018.8748992.

[3] S. Sivaranjani, S. Ananya, J. Aravinth and R. Karthika, "Diabetes Prediction using Machine Learning Algorithms with Feature Selection and Dimensionality Reduction," 2021 7th International Conference on Advanced Computing and Communication Systems (ICACCS), Coimbatore, India, 2021, pp. 141-146, doi: 10.1109/ICACCS51430.2021.9441935.

[4] A. C. Lyngdoh, N. A. Choudhury and S. Moulik, "Diabetes Disease Prediction Using Machine Learning Algorithms," 2020 IEEE-EMBS Conference on Biomedical Engineering and Sciences (IECBES), Langkawi Island, Malaysia, 2021, pp. 517-521, doi: 10.1109/IECBES48179.2021.9398759.

[5] R. Rastogi and M. Bansal, 'Diabetes prediction model using data mining techniques', Measurement: Sensors, vol. 25, p. 100605, Feb. 2023.

[6] S., Reshmi & Biswas, Saroj & Nath Boruah, Arpita & Thounaojam, Dalton & Purkayastha, Biswajit. (2022). Diabetes Prediction Using Machine Learning Analytics. 108-112. 10.1109/COM-IT-CON54601.2022.9850922.

[7] N. Mohan and V. Jain, "Performance Analysis of Support Vector Machine in Diabetes Prediction," 2020 4th International Conference on Electronics, Communication and Aerospace Technology (ICECA), Coimbatore, India, 2020, pp. 1-3, doi: 10.1109/ICECA49313.2020.9297411.

[8] Dey, S. K., Hossain, A., & Rahman, M. M. (2018). Implementation of a Web Application to Predict Diabetes Disease: An Approach Using Machine Learning Algorithm. 2018 21st International Conference of Computer and Information Technology (ICCIT).

[9] Alanazi, A. S., & Mezher, M. A. (2020). Using Machine Learning Algorithms For Prediction Of Diabetes Mellitus. 2020 International Conference on Computing and Information Technology (ICCIT-1441). doi:10.1109/iccit-144147971.2020.9213708.

[10] A. Mir and S. N. Dhage, "Diabetes Disease Prediction Using Machine Learning on Big Data of Healthcare," 2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA), Pune, India, 2018, pp. 1-6, doi: 10.1109/ICCUBEA.2018.8697439.

[11] P. M. S. Sai, G. Anuradha and V. P. kumar, "Survey on Type 2 Diabetes Prediction Using Machine Learning," 2020 Fourth International Conference on Computing Methodologies and Communication (ICCMC), Erode, India, 2020, pp. 770-775, doi: 10.1109/ICCMC48092.2020.ICCMC-000143.

[12] G. K. Teimoory and M. Reza Keyvanpour, "An Effective Feature Selection for Type II Diabetes Prediction," 2024 10th International Conference on Web Research (ICWR), Tehran, Iran, Islamic Republic

of, 2024, pp. 64-69, doi: 10.1109/ICWR61162.2024.10533371.

[13] A. R. J and R. Kotian, "Diabetes Prognosis using Machine Learning," 2022 4th International Conference on Inventive Research in Computing Applications (ICIRCA), Coimbatore, India, 2022, pp. 875-882, doi: 10.1109/ICIRCA54612.2022.9985487.

[14] Olaniyi, Ebenezer Obaloluwa, and Khashman Adnan. "Onset diabetes diagnosis using artificial neural network." Int J Sci Eng Res 5.10 (2014): 754-759.

[15] Sonar, Priyanka and K. Jaya Malini. “Diabetes Prediction Using Different Machine Learning Approaches.” 2019 3rd International Conference on Computing Methodologies and Communication (ICCMC) (2019): 367-371.