

DIABETES DIAGNOSIS USING MACHINE LEARNING

Jayapal,**Doddagouda R Patil**,***Sagar Srivastava***,****Kamil Firas khan****,

*****Dr.K. Ramesh. Associate Professor, Dept of EIE. RV College of Engineering.*****

Electronics And Instrumentation Engineering, R V College Of Engineering, Bengloure, India

ABSTRACT - Diabetes a chronic metabolic disorder, has a rich history dating back to ancient times. It was first documented in ancient Egyptian manuscripts around 1500 BCE. However, significant advancements occurred in the 20th century with the discovery of insulin in 1921, revolutionizing its treatment. Diabetes, a prevalent disease, requires frequent glucose level monitoring, typically through uncomfortable finger pricks. Manual calculations for insulin dosage add to the inconvenience. This project implements a smart program to emulate the pancreas' function, utilizing a machine learning-based model for real-time glucose tracking and accurate diabetes detection. According to data from National Family Health Survey (NFHS-5), the prevalence of diabetes mellitus (DM) and hypertension (HTN) in Karnataka is 14.8% and 25.9% respectively.

In This proposed project, microcontroller-based noninvasive blood glucose detection system is developed. In the proposed system temperature, oxygen and glucose sensor being used and the output of the sensor are fed to microcontroller for data processing through data acquisition system. The average and Gaussian filters are designed and develop to remove the noise. The code is developed using python for data processing. The microcontroller module receives the sensor data and is uploaded to the cloud platform. Microcontroller having its built-in Wi-Fi functionality, the establishes API communication with the thingspeak Iot platform. Acting as a client, the microcontroller sends the sensor data to the thingspeak server. Subsequently, thingspeak receives the sensor data from the cloud and presents the obtained results in a graphical format.

Thingspeak serves as an Iot platform that collects and stores sensor data. The data can be retrieved from thingspeak and preprocessed to prepare it for analysis using Random Forest algorithm implemented using Python. The Random Forest

algorithm is implemented using libraries like scikit-learn, is used for machine learning tasks such as classification. The data is split into training and testing sets, where the Random Forest model is trained on the training data. This model, consisting of an ensemble of decision trees, learns patterns and relationships in the data. The model's performance is then evaluated using the testing data. To display the results on web page, a web development framework like Flask or Django is used. A web application incorporating the trained Random Forest model is created, allowing users to input data through the web page. The model is invoked to make detection based on the user input, and the detected results are displayed on the webpage.

KEY WORD:- sensors, Node MCU, CLD display, Thingspeak Iot, machine learning algorithm.

INTRODUCTIN

Diabetes is a fast-growing disease among people even among youngsters. Diabetes is a disease caused due to the increase level of blood glucose. This high blood glucose produces the symptoms of frequent urination, increased thirst, and increased hunger. Diabetes is a one of the leading cause of blindness, kidney failure, amputations, heart failure and stroke. When we eat, our body turns food into sugars, or glucose. At that point, our pancreas is supposed to release insulin. Insulin serves as a key to open our cells, to allow the glucose to enter and allow us to use the glucose for energy. But with diabetes, this system does not work.

we are using Machine learning is a branch of artificial intelligence (AI) that focuses on building algorithms and models that enable machines to learn from data, identify patterns, and make predictions or decisions without being explicitly programmed

Machine learning algorithms can automatically improve their performance on a given task by

learning from examples, feedback, or experience. Machine learning has shown promising results in aiding the diagnosis of diabetes by analyzing patient data and identifying patterns that can indicate the presence of the disease.

The diagnosis of diabetes using machine learning involves building a model that can accurately classify patients into two categories: diabetic and non-diabetic. The model is trained on a dataset that includes patient information, such as age, gender, body mass index (BMI), blood pressure, and blood sugar levels, among others. Supervised learning algorithms, such as logistic regression, decision trees, and support vector machines (SVMs), can be used to build the classification model. The model is trained on a labeled dataset, where each patient is assigned a label indicating whether they are diabetic or not.

Motivation

Diabetes is a chronic metabolic disorder that affects millions of people worldwide and can lead to serious health complications if left untreated. Early detection and timely intervention can significantly improve patient outcomes and reduce the risk of complications. The use of machine learning in diabetes diagnosis can also help healthcare providers to optimize their resources by identifying high-risk patients, reducing the need for unnecessary testing, and improving patient outcomes. Overall, this project can have a significant impact on the quality of care for patients with diabetes and help to reduce the burden of this disease on healthcare systems.

PROBLEM STATEMENT

Design and develop a device that can diagnose Diabetes using Machine Learning algorithms.

DATA SET DETAILS

- Glucose: Plasma glucose concentration a 2hours in an oral glucose tolerance test
- Blood Pressure: Diastolic blood pressure (mm Hg)
- Skin Thickness: Triceps skin fold thickness, (mm)
- Insulin: 2-Hour serum insulin (mu U/ml)

- BMI: Body mass index (weight in kg/(height in m)²)
- Diabetes Pedigree Function: Diabetes pedigree function
- Age: Age (years)
- Outcome: Class variable (0 or 1)

METHODOLOGY

The proposed system consists of 3 different phases data collection, storage and analysis. This approach places an important role in diagnosis diabetes and pre-training diabetes. The output is the accuracy metrics of the machine learning models. Then, the model can be used in diagnosis. Algorithm used are Linear Regression, Logistic Regression and Decision-tree-model.

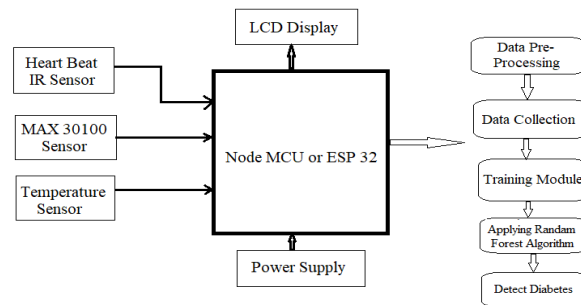


Fig 1.1_Block diagram.

A block diagram is a visual show that methodology is.

- 1) Gather the components: Ensure that all the necessary components are available, including the sensors, ESP32 microcontroller, LCD display, and a suitable power supply.
- 2) Gather the components: Ensure that all the necessary components are available, including the sensors, ESP32 microcontroller, LCD display, and a suitable power supply.
- 3) Power supply: Connect the power supply to the ESP32. The power supply options may include a battery, a USB cable connected to a power source, or any other appropriate power source. Make certain that the power supply voltage aligns with the requirements of the ESP32 and other components.
- 4) Sensor connections: Establish specific connections between each sensor and the ESP32. Typically, sensors employ digital protocols such as I2C, SPI,

or UART for communication.

- 5) ESP32 connections: Connect the relevant communication pins of each sensor to the corresponding pins on the ESP32.
- 6) LCD display connection: Similarly, establish the necessary connections between the LCD display and the ESP32. The specific connections will depend on the type of LCD display being used.
- 7) Code development: Write or obtain the requisite code to read data from the sensors and display it on the LCD. Utilize suitable libraries and programming languages for the ESP32 platform. The code should encompass sensor initialization, data reading, and data transmission to the LCD display.
- 8) Upload code: Compile the code and upload it to the ESP32 using the Arduino IDE or any other appropriate programming environment for ESP32 development.
- 9) Testing: Power on the system once the code is uploaded and ensure that the sensors are correctly connected and operational. Verify that the sensor data is accurately displayed on the LCD display.
- 10) ThingSpeak IoT Platform: The Node MCU/ESP32 sends the collected data (heart rate, temperature, glucose, SPO2) to the ThingSpeak IoT platform.
- 11) Web Page (Graphical Representation): The ThingSpeak platform generates graphical representations of the collected sensor data, allowing users to view and analyze the data through a web page interface.
- 12) Machine Learning (Python Coding): The collected sensor data can be retrieved from ThingSpeak using Python code. Various methodologies can be used for collecting the data and performing the analysis. Some common approaches include:
 - a. Data Preprocessing: The sensor data may require preprocessing steps such as normalization, outlier removal, or feature extraction before feeding it into the machine learning algorithm.
 - b. Data Split: The data is split into a training set and a testing set. The training set is used to train the machine learning model, while the testing set is used to evaluate its performance.
 - c. Machine Learning Algorithm: Different algorithms can be used for diabetic diagnosis, such as Random Forest, Support Vector Machines, or Neural Networks. The choice of algorithm depends on the specific requirements and characteristics of the

data.

- d. Model Training: The machine learning model is trained using the training data. The model learns patterns and relationships in the data to make predictions.
- e. Model Evaluation: The trained model is evaluated using the testing data to assess its accuracy and performance in detecting diabetes.
- 13) Diabetic Diagnosis Web Page Result: The result of the diabetic diagnosis, based on the machine learning model, is displayed on a web page for users to view.

Please note that this block diagram provides a general overview of the workflow for your project. The specific implementation details may vary based on your project requirements, the hardware and software choices, and the specific tools and frameworks you use.

EXISTING SYSTEM

Existing machine learning models and their comparison with the proposed work is given in Table 3. Implementation of SVM and other algorithms results in a best accuracy of 77.73% in this paper [8]. Similarly, Naïve Bayes has an accuracy of 76.34% in [11], LR while comparing other classifiers like KNN, DT, RF etc. is efficient with 77.64% in [12] and among SVM, DT, RF, LR and KNN, RF was better with 74.44% accuracy. In our proposed work, the model selects four most contributing features and the accuracy after step forward feature selection in RF is 83% and 80% in SVM classifiers.

A. HARDWARE REQUIREMENTS

- Processor : >i3
- Ram :4GB.
- HardDisk : 500GB.
- Inputdevice : Standard Keyboard andMouse.
- CompactDisk : 650Mb.
- Outputdevice : High ResolutionMonitor.
- Max 30102 heart beat sensor

B. SOFTWARE REQUIREMENTS

- Operating system : macOS, Windows

XP/7 or higher version.

- Coding language : python ($\geq$ python 3.3 or python2.7).
- IDE : Jupyter Notebook.

ARCHITECTURE DIAGRAM

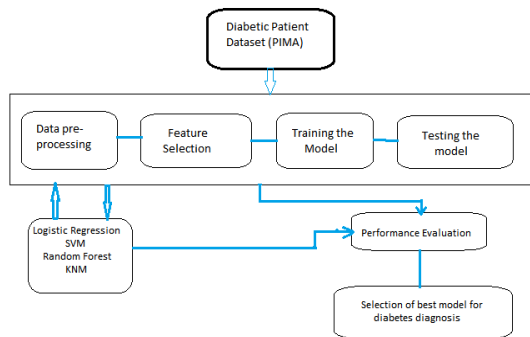


Fig.1.2 Architecture Diagram

IMPLEMENTATION

Dataset: The data set which is used in this project has been taken from Pima Indians diabetes database which is of National institute of diabetes and digestive and kidney diseases. The most important purpose of the dataset used is to verify with diagnostically to predict if patient has diabetes or not based only on the defined diagnostic measurements which includes in the database. The data of all the patients in this particular data set are women at least 21 years old of Pima Indian heritage. The outcome is the dataset in medical predictor and consists only one target variable. As shown in the figures they are in depended to each other: Based on the first 7 independent column values, we are going to predict our model which is machine learning model and then will predict the value of the last column, that is the outcome. 1 and 0 are the two medical values considered here. In that 1 indicates that the patient is diabetic and 0 indicates person is not diabetic.

To carry out diabetes prediction, we have taken the existing data set which is from Pima Indian Dataset from Kaggle. This particular data set is taken from the National Institute of Diabetes and Digestive and Kidney Diseases. And then data processing is performed and the data being used will be divided into two sets which are training set and testing set. Now they are sent to ML model which is machine learning model where we have used algorithms like Logistic Regression, KNN, Random Forest classifier, SVM. We have selected random forest classifier based on train accuracy and test accuracy which was high compared to all other

algorithms which has testing accuracy 93% and training accuracy of 98 %.

Web page creation is done where the user can give user inputs to get the results. The factors to enter in the web page are:

- | | |
|--------------------|------------------|
| -Pregnancies | -Glucose |
| - BMI | -Insulin |
| -Diabetes pedigree | -Blood pressure |
| -Age | - Skin thickness |

Steps in implementation: 1. Training the model using the given dataset. 2. Install the necessary software's like PyCharm IDE, Django, Anaconda and Web browser. 3. Basic setting on PyCharm 4. Design website home page (Frontend) 5. Design the web page for user input and prediction. 6. Link the trained model to the front end.

EXPECTED OUTCOME

To develop a system which can perform diagnosis of diabetes for a patient with a higher accuracy by combining the results of different machine learning techniques

CONCLUSION

The project on "Using Sensors to Detect Diabetes Diagnosis" has successfully demonstrated the practical application of sensor technology in detecting diabetes through the integration of a machine learning algorithm. By leveraging sensors for temperature, heartbeat, glucose, and oxygen levels, real-time data from the body was collected and processed. The machine learning algorithm, specifically tailored for diabetes detection, analysed the data and provided accurate predictions for diagnosis. This project showcases the potential of sensor technology and machine learning in healthcare, specifically in the early detection and monitoring of diabetes. The successful implementation of this system highlights the advancements made in medical diagnostics, contributing to improved healthcare outcomes and better quality of life for individuals with diabetes. Future research and development in this field hold great promise for further refining and expanding the capabilities of sensor-based diabetes detection systems, leading to even more effective diagnostic tools in the future.

REFERENCES

- [1] “Random Forests for Diabetes Diagnosis” by Sofia Benbelkacem and Baghdad Atmani, Laboratoire d’Informatique d’Oran (LIO) University of Oran 1 Ahmed Benbella Oran, Algeria. April 2019 International Conference on Computer and Information Sciences (ICCIS)
- [2] “Classification of Diabetes Disease Using Support Vector Machine” by V. Anuja Kumari, R. Chitra Noorul Islam university. ISSN: 2248- 9622 www.ijera.com Vol. 3, Issue 2, March - April 2013, pp.1797-1801
- [3] “Diabetes Prediction Based on XGBoost Algorithm” by Mingqi Li, Xiaoyang Fu and Dongdong Li. 2020 IOP Conf. Ser.: Mater. Sci. Eng. 768 072093
- [4] “Diagnosis of Diabetes Mellitus using K Nearest Neighbor Algorithm” by Krati Saxena, Dr. Zubair Khan, Shefali Singh. Volume 2 Issue 4, July-Aug 2014
- [5] “Diabetics Prediction using Gradient Boosted Classifier” by J. Beschi Raja, R. Anitha, R.Sujatha, V. Roopa, S. Sam Peter. ISSN: 2249 – 8958, Volume-9 Issue-1, October 2019
- [6] “Prediction of Diabetes by Using Artificial Neural Network, Logistic Regression Statistical Model and Combination of Them” by Paratoo RAHIMLOO, Ahmad JAFARIAN. Vol. 85, 2016, p. 1148 – 1164
- [7] “Analysis and Detection of Diabetes using Data Mining Techniques- A Big Data application in health care” by B. G. Mamatha Bai, B. M. Nalini and Jharna Majumdar. Nitte Meenakshi Institute of Technology, January 2019