FACULTY OF ENGINEERING AND TECHNOLOGY

DEPARTMENT OF COMPUTER ENGINEERING

Artificial Intelligence - ENCS434

Diabetes prediction

Prepared by: Leyan Burait #1211439

Haneen Odeh#1210716

Instructor: Dr. Adnan Yahya

Section: 1

Abstract:

Diabetes is a complex chronic disease that requires accurate classification and prompt diagnosis for effective management. This project aims to develop a robust classification and diagnostic system for diabetes using machine learning techniques and the Python language in the Virtual Studio application, as well as the Weka application. We utilize a diverse dataset containing clinical and laboratory features, preprocess and transform the data to extract meaningful information, and train and optimize our classification model using various machine learning algorithms available in the Weka program, such as decision trees, neural networks, and naive Bayes. By evaluating the model's performance metrics, including accuracy, precision, recall, and F1-score, we demonstrate the effectiveness of our system in predicting diabetes onset and categorizing patients into relevant diabetes types. This implementation using both the Virtual Studio and Weka applications provides a comprehensive and efficient tool for healthcare professionals to aid in early diabetes detection and personalized treatment strategies

Table of Contents

1. Methodology

we took to complete our project to detect diabetes We followed the following steps:

1. We first read the raw data and performed all the preprocessing
2. All pre-processed data was rewritten with the extracted data into a CSV file.
3. Next, we decided which model we would practice on, and made sure we had a cleaned data and then train the model
4. We stored all trained models for future use and new test samples.

1.1.Test

we will depend on a processed data entries stored on a csv file we use 2769 processed data entries stored Before going in depth with the different outputs, we need to consider the following metrics:

1. Precision: is the number of correctly classified positive examples divided by the total number of examples that are classified as positive.

2. Recall: is the number of correctly classified positive examples divided by the total number of actual positive examples in the test set.

3. Accuracy: is the number of correct classification divided by the total number of test cases.

4. F1 score: is the harmonic mean of precision and recall, and it tends to closer the smaller of the two.

1.2.Python language in the Virtual Studio application:

By following these steps, users can effectively train, optimize, and evaluate their classification models using decision trees, neural networks, naive Bayes, and random forest by python in the virtual studio. The assessment of accuracy, precision, recall, and F1-score serves as evidence to demonstrate the effectiveness of the developed system in predicting diabetes and classifying patients into relevant diabetes types.

1. Data Preprocessing: Start by preprocessing the dataset containing clinical and laboratory features. This may involve handling missing values, scaling numerical features, and encoding categorical variables. 2. Feature Extraction: Extract relevant features from the preprocessed data that are likely to contribute to diabetes classification. This can involve techniques such as principal component analysis (PCA) or feature selection algorithms.

3. Splitting Data: Split the preprocessed data into training and testing datasets. The training dataset will be used to train the machine learning models & we use 80% of the dataset , while the testing dataset will be used to evaluate their performance & we use 20% of the dataset.

4. Model Training: Train the classification models using machine learning algorithms like decision trees, neural networks, naive Bayes, and random forest. These algorithms will learn patterns from the training data and create models that can classify future instances.

5. Model Evaluation: Evaluate the trained models using various evaluation metrics like accuracy, precision, recall, and F1-score. These metrics provide insights into the model's performance and its ability to predict diabetes onset and categorize patients correctly.

We use extra a logarithm Random Forest: The Random Forest algorithm is a versatile and widely used machine learning technique that operates by constructing multiple decision trees during training and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees.

2. Procedure

2.1 Models Training

In this project, various machine learning techniques, including decision trees, naïve byes and neural networks, are utilized to train and optimize a classification model by python language and its packages, before we use the test & training dates we cleaned them to be ready for usages we use the three algorithm below to training- & testing the dataset.

A Decision Tree: algorithm is employed as a powerful tool for constructing a tree-like model of decisions and their possible consequences.

A Naive Bayes classifier: It is a probabilistic machine learning model that's used for classification task. It is based on the Bayes theorem

A Neural Network: a type of deep learning algorithm, are also employed in this project.

Random Forest: algorithm is a versatile and widely used machine learning technique that operates by constructing multiple decision trees during training and outputting the class that is the mode of the classes (classification) or mean prediction (regression) of the individual trees.

2.2 Datasets

In our project we have used 3 different datasets, so we can test and check , each dataset is saved as .csv file

1) first dataset of 2768 processed

we use it in Weka & in we work in virtual studio Link: diabetes dataset. Kaggle. https://www.kaggle.com/datasets/nanditapore/healthcare-diabetes

2) second dataset of 768 processed

we use in we work in virtual studio Link: Diabetes. Kaggle. https://www.kaggle.com/datasets/salihacur/diabetes

3) third dataset of 4304 processed

We use in we work in virtual studio Diabetes_dataset_with_18_features. Kaggle.

[Diabetes_Dataset_With_18_Features (kaggle.com)]

2.3 Important packages

In our project, we have to installed packages so we can handle the machine learning algorithms we used, those packages are

main.py > ...

```python
#! Installing all packages needed
from sklearn.tree import DecisionTreeClassifier
from sklearn.naive_bayes import GaussianNB
from sklearn.neural_network import MLPClassifier
from sklearn.ensemble import RandomForestClassifier
from sklearn.model_selection import train_test_split
from sklearn.metrics import accuracy_score, precision_score, recall_score, f1_score
from sklearn.metrics import confusion_matrix
import pandas as pd
import time
```

2.4 The implementation

The main important thing in our project, is that letting the user decide which dataset he wants to perform, also which algorithm of the machine learning algorithms, so this menu will be displayed

```python

while True:
    #! Ask the user to choose the algorithm
    print("\n\t")
    print("Choose the algorithm you want to test:")
    print("1. Decision Tree")
    print("2. Naive Bayes")
    print("3. Neural Network")
    print("4. Random Forest")
    print("5. Exit")
    choice = input("Enter your choice (1, 2, 3, 4, or 5): ")

```

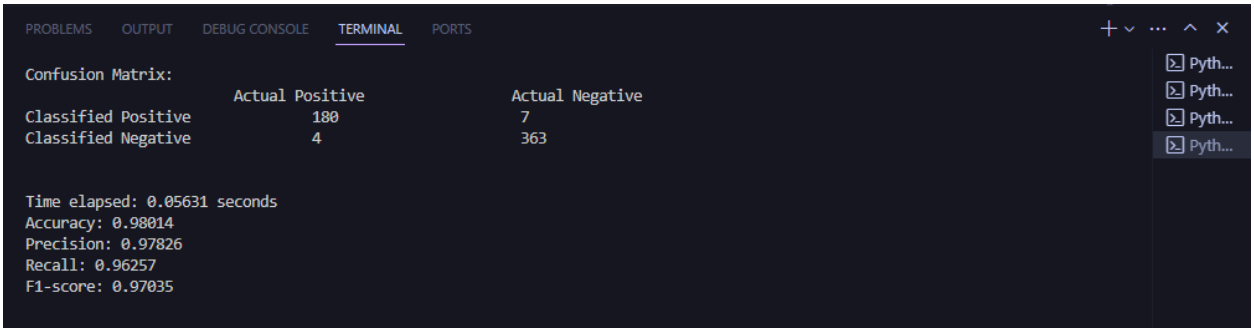
```python
#! Letting the user decide which dataset he/she wants
print("Last ID for dataset1:2768")
print("Last ID for dataset2:768")
print("Last ID for dataset3:4303, Note:(This dataset has more number of features)")

dataset_choice = input("Enter the dataset you want to test (1 or 2 or 3): ")

if dataset_choice == '1':
    data = data1
elif dataset_choice == '2':
    data = data2
elif dataset_choice == '3':
    data = data3
else:
    print("Invalid dataset choice.")
    exit()
```

2.5 Results and Outputs

For any algorithm the user choose, the program will calculate 4 main things, accuracy, precision , recall and F1-score, it will also display the confusion matrix for the user



2.6 Enter your data

As a bonus step, we implemented our program to take a data from the user, and test it to see if the diabetes is predicted or not