ENHANCING DIABETES DIAGNOSIS ACCURACY WITH AN ENSEMBLE LEARNING APPROACH

A DISSERTATION

SUBMITTED IN PARTIAL FULFILMENT OF THE REQUIREMENTS

FOR THE AWARD OF THE DEGREE

OF

MASTER OF TECHNOLOGY

IN

INFORMATION SYSTEMS

Submitted by:

DEEPANSHU

2K21/ISY/06

Under the supervision of

Dr. Varsha Sisaudia
Assistant Professor

Department of Information Technology

INFORMATION TECHNOLOGY

DELHI TECHNOLOGY UNIVERSITY

(Formerly Delhi College of Engineering)

Bawana Road, Delhi-110042

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Bawana Road, Delhi-110042

CANDIDATE'S DECLARATION

I Deepanshu 2K21/ISY/06 student of M.Tech INFORMATION SYSTEMS, hereby declare that the Thesis Dissertation titled "Enhancing diabetes diagnosis accuracy with an ensemble learning approach" which is submitted by me to the INFORMATION TECHNOLOGY, Delhi Technological University, Delhi in partial fulfillment of the requirement for the award of the degree of Master of Technology, is original and not copied from any source without proper citation. This work has not previously formed the basis for the award of any Degree, Diploma Associateship, Fellowship or other similar title or recognition.

Place: Delhi **DEEPANSHU**
Date:

INFORMATION TECHNOLOGY

DELHI TECHNOLOGICAL UNIVERSITY

(Formerly Delhi College of Engineering)

Bawana Road, Delhi-110042

CERTIFICATE

I hereby certify that the Thesis Dissertation titled "Enhancing diabetes diagnosis accuracy with an ensemble learning approach" which is submitted by Deepanshu 2K21/ISY/06 INFORMATION TECHNOLOGY, Delhi Technological University, Delhi in partial fulfillment of the requirement for the award of the degree of Master of Technology, is a record of the project work carried out by the students under my supervision. To the best of my knowledge his work has not been submitted in part or full for any Degree or Diploma to this University or elsewhere.

Place: Delhi

Date:

Dr. Varsha Sisaudia

SUPERVISOR

Assistant Professor

Abstract

The goal of this thesis is to evaluate how different machine learning algorithms perform in detecting diabetes using the Pima dataset. Three models are examined for their accuracy: Random Forest (72%), Long-Short-Term-Memory Networks (LSTMs) (75.32%), & Extreme-Learning-Machine (ELM) (77.70%). The findings demonstrate that the ensemble model containing ELM, LSTM, and RF outperforms the individual models with an accuracy of 93%. To strengthen the ensemble model further, a unique strategy is proposed, comprising a mix of ELM and LSTM models together with the addition of a Random Forest classifier. Cross-validation is utilised to test the suggested model, indicating its improved performance compared to the individual models, with a mean AUC of 0.93. Subsequently, the model is trained on the complete dataset and achieves an accuracy of 93% and an AUC of 0.89 on the testing set. These results show that the proposed approach efficiently diagnoses diabetes and offers promise for assisting clinical decision-making.

Acknowledgements

I am very thankful to Dr. Varsha Sisaudia, my thesis adviser, for her great advice, support, and encouragement during this journey. Her exceptional leadership and relentless passion have been the cornerstone of my academic achievement. Her insightful input, constructive criticism, and constant dedication to my progress have been priceless, and I could not have accomplished this milestone without her direction.

Dr. Sisaudia has been an amazing adviser, constantly going the additional mile to ensure that I got the tools and support required to succeed. Her willingness to share her expertise and experience has helped me grow my own abilities and confidence as a researcher. Her leadership, passion, and relentless work ethic have been a continual source of inspiration for me. I am very thankful for her consistent support and mentoring, and I will always treasure the information and skills that I have received under her supervision.

Table of Contents

List of Tables

Chapter 1

Introduction

Diabetes is a major chronic ailment that influences a large number of folks worldwide. It is characterized by high blood sugar level, contributing to diverse complications like heart disease, stroke, kidney disease, and vision impairment. Timely detection of diabetes is critical for properly controlling and avoiding these consequences. Machine learning algorithms have demonstrated promising effects in identifying diabetes utilising varied datasets.

This work concentrates on the Pima Indians Diabetes Dataset, which comprises several physiological and diagnostic data for predicting diabetes. Four unique machine learning methods, namely Random Forest, LongShort-TermMemory Network(LSTMs), ExtremeLearningMachine (ELM), and an ensemble model incorporating ELM, LSTM, and Random Forest, have been utilised to predict diabetes on this dataset.

Furthermore, we offer a new ensemble model that integrates ELM and LSTM models to boost the accuracy of diabetes prediction. The performance of recommended ensemble model will be tested with several machine learning models to discover the most effective way for detecting diabetes in the Pima dataset.

The suggested ensemble model consolidates the predictions from the ELM and LSTM models and employs them as input for a Random Forest-model. CrossValidation will be done to tweak the hyperparameters of the Random Forest model and analyse the final models performance on the testing set.

The primary goal of this work is to examine the performance of multiple machine learning(ML) approaches and propose a novel ensemble model for diagnosing diabetes in the Pima Diabetes Dataset.

Chapter 2

Literature review

1. The number of individuals living with diabetes globally is expanding quickly, becoming a global hazard and a primary cause of mortality. However, diabetes is mainly avoidable with lifestyle modifications. To help in early identification and prescribe essential lifestyle alterations, there is a need for a prognostic tool. Machine learning algorithms may aid in predicting diabetes by evaluating massive healthcare datasets. Among many classifiers, Deep-Learning (DL) has showed the greatest accuracy of 98.7% in predicting diabetes onset using the PIMA dataset. This study offers an excellent tool for healthcare practitioners and may help to the development of automated systems for early illness identification. Further advances may be achieved by combining omics data into the prediction process [1].

2. Based on lifestyle and family history, this research seeks to determine the likelihood of developing Type 2 diabetes in India. Machine learning techniques are utilised to properly estimate diabetes risk, and a dataset of 952 examples is obtained for testing. The Random-Forest-Classifier demonstrates maximum accuracy for both the obtained dataset and the Pima Indian Diabetes database. The results underscore the necessity of early diagnosis of diabetes risk and give useful insights into critical factors for prediction. Further study may examine other machine learning techniques for improved illness prediction [2].

3. This paper tackles the detection of diabetes using the Pima dataset. It analyses common approaches like DNN and SVM, together with data pretreatment methods, to achieve accurate identification. The paper performs detailed assessments using cross-validation and analyses the accuracy of several classifiers with various data preprocessors and parameter tweaks. The best method yields a 77.86% accuracy using 10 fold cross validation. The research

also explores the association between each attribute and the categorization result [3].

4. Diabetes is a prevalent metabolic illness that needs early identification to avert complications. This research makes use of machine learning(ML) methods to examine Pima dataset to uncover patterns and risk factors related with diabetes. Five supervised ML methods, including SVM-linear, RBF kernel SVM, KNN, ANN, and MDR, are utilised to categorise patients as diabetic or non-diabetic. The work intends to increase prediction accuracy and contribute to the knowledge of diabetes-related variables utilising ML and data manipulation tools in R [4].

5. The relevance of illness prediction in medical diagnostic software and the effective implementation of machine learning methods in different domains, including healthcare. By incorporating machine learning algorithms and a classifier system, clinicians may enhance illness prediction and early-stage diagnosis. The study focuses on diabetes detection using machine learning(ML) techniques and the PIMA dataset. Various approaches, including ANN,RF , NBayes, KNN, SVM, and LR, are used, and their results, as well as their advantages and disadvantages, are reviewed [5].

6. This research provides a modified Support-Vector-Machine (SVM) approach for diagnosing Pima Dataset. The suggested technique surpasses prior classification methods, including Neural-Network, RBFNetworks, and KNN. The approach separates the training data into two sections and trains distinct SVM models using different kernels. During classification, the algorithm decides the SVM model to apply depending on the dataset's properties. The proposed technique had an average classification success rate of 82.2%, outperforming previous research's best result [6].

7. Diabetes is a major health risk for American Indians in the United States. The UCI Machine Learning Lab's Pima Indian diabetic database (PIDD) is extensively used to predict diabetes status. In this research, an ensemble model integrating Support-Vector-Machine(SVM) and Backpropagation-Neural-Network(BPNN) was constructed to predict the existence of diabetes. The ensemble model shown an outstanding predicted accuracy of 88.04%,

outperforming previous categorization methods in the literature. This study indicates the promise of the ensemble technique in correctly predicting diabetes, leading to early intervention and better healthcare outcomes for American Natives [7].

8. Gestational diabetes is a prevalent condition among Indian pregnant women, with possible hazards to the kid if left untreated. Early detection is vital to avoid problems. The Decision Tree J48 method is recommended in this research for predicting gestational diabetes. The dataset includes 768 patient records with eight variables and a target column representing diabetes status. The experiment reveals that the Decision Tree J48 method gives efficient and accurate predictions with decreased processing time. This study helps to advancing understanding and improving healthcare for gestational diabetes [8].

9. Deep Learning has emerged as a useful tool in the realm of medical applications, including diabetes categorization. This research focuses on utilising Multi-Layer-Feed-Forward Neural-Networks (MLFNN) to categorise diabetes in the Pima Indian Diabetes dataset. The study tackles numerous strategies, such as activation functions, learning algorithms, and managing missing variables, to boost classification accuracy. The findings are compared to Naive Bayes and Random Forest algorithms, with MLFNN attaining the maximum accuracy of 84.17%. This illustrates the potential of Deep Learning in boosting diabetes classification accuracy compared to standard machine learning techniques [9].

10. This study focuses on employing machine learning approaches to enhance the early identification and diagnosis of diabetes. Two datasets, including a diabetic clinical dataset and the PIMA Indian diabetic dataset, are studied. Various classifiers and activation functions are investigated, with logistic regression finding to produce the greatest performance in categorising diabetes and non-diabetic individuals. The research demonstrates the use of machine learning(ML) in improving diabetes diagnosis and emphasises the effectiveness of logistic regression in this context [10].

11. This research presents an ensemble hierarchical model for enhancing classification accuracy in diabetes prediction. The model combines separately trained Decision-Tree(DT) and Logistic-Regression(LR) classifiers and feeds their outputs into a Neural-Network(NN) for further refinement. The suggested strategy is assessed on the PIMA dataset and obtains a classification accuracy over 83%, exceeding current state of the art approaches. The hierarchical nature of the model enables for collecting varied patterns and information in the data, leading to more accurate predictions. This study makes a contribution to the field of machine learning(ML) and provides insights for better diabetes diagnosis and treatment [11].

12. Diabetes is a common condition impacting millions of individuals globally, especially women. Healthcare studies have embraced sophisticated technologies like machine learning to better diagnosis and prediction based on clinical data. This research focuses on predicting diabetes in Pima Indian women using a binary classification technique. Various supervised learning methods, including Logistic Regression, are applied and assessed using performance measures such as AUC, classification accuracy, F1-score, precision, and recall are all important metrics to consider. The results show that Logistic-Regression outperforms other algorithms in predicting diabetes in this group [12].

13. This paper describes an integrated method for diabetes prediction that combines the Synthetic-MinorityOversampling-Technique(SMOTE) and Sequential-MinimalOptimization(SMO) algorithms. On the PIMA dataset, the approach achieved a high accuracy rate of 99.7%. The findings indicate that this integrated strategy might operate as an expert system for diabetes detection. The research also highlighted the potential utility of the findings features in developing a smartphone app for early diabetes detection [13].

14. This work focuses on diabetes prediction utilising machine learning techniques and a pipeline model. The suggested strategy enhances classification accuracy relative to current datasets, with Logistic Regression obtaining 96% accuracy and the AdaBoost classifier in the pipeline model achieving 98.8% accuracy. The research finds that the model boosts diabetes

prediction accuracy and advises future examination of predicting diabetes risk in non-diabetic persons [14].

Chapter 3

Dataset

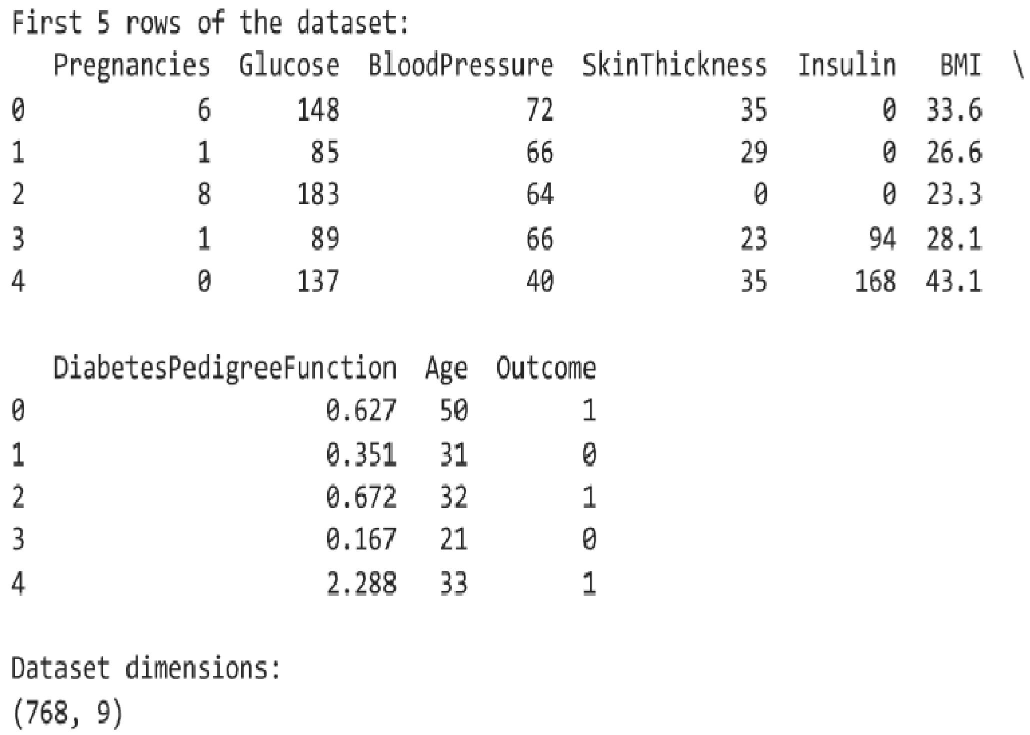
The pandas package is used to load the Pima Diabetes dataset. The dataset is then described in a variety of ways. To begin, the code displays the first 5 rows of the dataset, giving you a quick overview of the data structure. Following that, it displays the dataset's dimensions, including the number of rows and columns.

Then it continues to generate and show the statistical summary of the dataset, which includes measurements such as mean, standard deviation, minimum, maximum, and quartiles for each numerical characteristic. This summary provides information on the dataset's distribution and range of values.

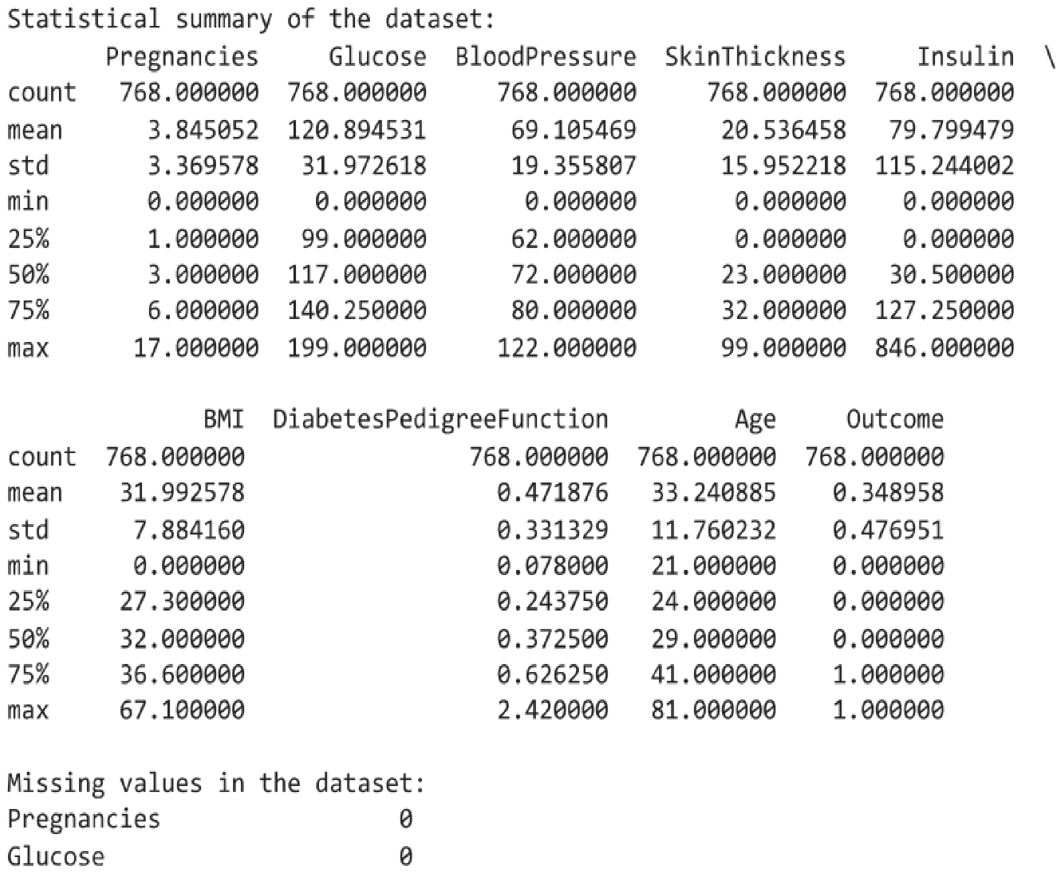
To assure data quality, the code checks for missing values in the dataset by counting the proportion of null values in each column. This stage is critical for discovering any gaps or anomalies in the data.

Lastly, the algorithm evaluates the class distribution of the 'Outcome' variable, which reflects the existence or absence of diabetes. It counts the number of instances in each class, offering an insight of the imbalance or balance between the classes.

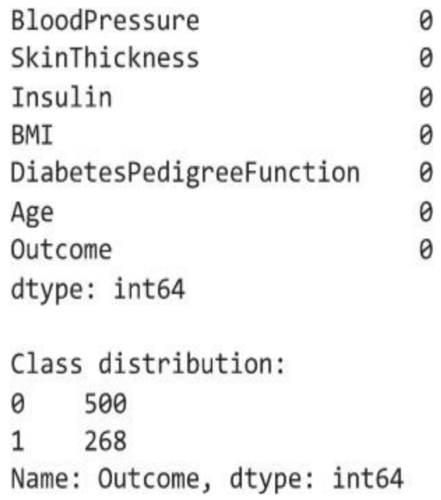
Overall, this code displays the first exploratory data analysis of the Pima dataset, giving vital information for additional analysis and modeling in the thesis.



```
First 5 rows of the dataset:
   Pregnancies  Glucose  BloodPressure  SkinThickness  Insulin   BMI  \
0            6      148             72             35        0  33.6
1            1       85             66             29        0  26.6
2            8      183             64              0        0  23.3
3            1       89             66             23       94  28.1
4            0      137             40             35      168  43.1

   DiabetesPedigreeFunction  Age  Outcome
0                     0.627   50        1
1                     0.351   31        0
2                     0.672   32        1
3                     0.167   21        0
4                     2.288   33        1

Dataset dimensions:
(768, 9)
```



```
Statistical summary of the dataset:
       Pregnancies     Glucose  BloodPressure  SkinThickness     Insulin  \
count   768.000000  768.000000     768.000000     768.000000  768.000000
mean      3.845052  120.894531      69.105469      20.536458   79.799479
std       3.369578   31.972618      19.355807      15.952218  115.244002
min       0.000000    0.000000       0.000000       0.000000    0.000000
25%       1.000000   99.000000      62.000000       0.000000    0.000000
50%       3.000000  117.000000      72.000000      23.000000   30.500000
75%       6.000000  140.250000      80.000000      32.000000  127.250000
max      17.000000  199.000000     122.000000      99.000000  846.000000

              BMI  DiabetesPedigreeFunction         Age     Outcome
count  768.000000                768.000000  768.000000  768.000000
mean    31.992578                  0.471876   33.240885    0.348958
std      7.884160                  0.331329   11.760232    0.476951
min      0.000000                  0.078000   21.000000    0.000000
25%     27.300000                  0.243750   24.000000    0.000000
50%     32.000000                  0.372500   29.000000    0.000000
75%     36.600000                  0.626250   41.000000    1.000000
max     67.100000                  2.420000   81.000000    1.000000

Missing values in the dataset:
Pregnancies                 0
Glucose                     0
```



```
BloodPressure               0
SkinThickness               0
Insulin                     0
BMI                         0
DiabetesPedigreeFunction    0
Age                         0
Outcome                     0
dtype: int64

Class distribution:
0    500
1    268
Name: Outcome, dtype: int64
```

Chapter 4

Methodology

Proposed Ensemble Model

1. Data Loading and Preprocessing:

The Pima diabetes dataset is imported from a remote URL into the programme using the pandas library. The input features are separated from the target variable (class) and assigned to the variables X and y to prepare the dataset for future analysis. The dataset is then divided into training and testing sets with a 9:1 ratio using the train-test-split function.

2. Training ELM model:

A model based on the Extreme-Learning-Machine(ELM) is developed using the hpelm library. The train function is then used to train the ELM model on the training set.

3. Training LSTM model:

An LSTM(Long Short Term Memory) model is created and trained on the training set using the Keras library. The input data is moulded into a format (samples, timesteps, features) that meets the LSTM model's requirements. The fit function is used in the training method, using binary_crossentropy as the loss function, the AdamOptimizer, and accuracy as the assessment metric.

4. Ensemble of ELM and LSTM models:

Predictions from the ELM and LSTM models are concatenated using numpy concatenate function.

5. Training Random Forest model:

On the concatenated predictions, the fit function is utilised to train a Random-Forestclassifier. The hyperparameters n-estimators and max-depth are specified as 500 and 5, respectively. To test the model's performance, cross-validation is undertaken using the cross_val_score function, which provides the mean AUC score.

6. Assessing the Random Forest model's performance using metrics and visuals:

Using the trained Random Forest model, the testing set's class labels are predicted, as well as the accuracy and AUC score of the predictions are obtained using the accuracy_score and roc_auc_score functions, respectively.

Extreme Learning Machine (ELM)

1. Utilize the pandas library to load the Pima dataset.

2. Distinguish between the input attributes X and the target variable y.

3. Remove the mean and divide by the standard deviation to normalise the input characteristics.

4. Create a function named elm accepts input features, target variable, number of hidden neurons, and activation function as arguments.

5. Randomly initialize the weight and biases of the input to hidden layer.

6. Apply the specified activation function to calculate the output of the hidden layer.

7. Identify the concealed layer's Moore-Penrose pseudoinverse.

8. Compute output weights using the pseudoinverse.

9. Compute the predictions by multiplying the hidden layer output with the output weights.

10. Convert the predictions to class labels by rounding them to the nearest integer.

11. Evaluate the model accuracy by computing the mean of the binary accuracy for each sample.

12. Test the elm function with 20 hidden neurons using both sigmoid and tanh activation functions.

Long Short Term Memory Networks (LSTMs)

1. Import necessary libraries including pandas, numpy, tensorflow, sklearn.model_selection, and sklearn.preprocessing.

2. In a pandas DataFrame, load the Pima diabetes dataset.

3. Distinguish the input attributes (X) from the target variable (y) in the dataset.

4. Utilise the train_test_split method in sklearn.model_selection to split the dataset into training and testing sets.

5. Apply feature scaling to the input features using the StandardScaler() method from sklearn.preprocessing.

6. Reshape the input features to fit the predicted input shape of the LSTM model.

7. Define the LSTM model using the Sequential() method from tf.keras, which allows for stacking layers consecutively.

8. Compile the LSTM model using the compile() function, choose binary_crossentropy as the loss function, Adam as the optimizer, and accuracy as the evaluation measure.

9. Using the fit() function, train the LSTM model using the inputs X_train and y_train, the batch size set to 32, the number of epochs to 50, and the validation data specified as (X_test, y_test).

10. Evaluate the trained LSTM model using the evaluate() function, using X_test and y_test as inputs, and display the accuracy score.

Random Forest

1. The relevant libraries are imported at the beginning of the code, including pandas, train_test_split, RandomForestClassifier, and accuracy_score.

2. The Pima Indian diabetes dataset is imported into the variable 'pima_data' using the pandas library.

3. The input characteristics (X) and target variable (y) are retrieved from the dataset.

4. The train_test_split function from the sklearn package is used to divide the dataset into training and testing sets.

5. A Random-Forest classifier with 100 trees is created by setting the n_estimators parameter to 100.

6. The Random-Forest-classifier is trained on the training data using the fit() method.

7. Using the Random-Forest-classifier's predict() method, predictions are made on the testing data.

8. The accuracy_score() function from the sklearn.metrics package is used to compare the predicted and actual labels and calculate the model's accuracy.

9. The Random Forest model's validity is printed.

Chapter 5

Data Analysis

Proposed Ensemble Model

1. Libraries Imported:

The required libraries for this analysis are imported - numpy, pandas, sklearn, hpelm, and keras.

2. Data Loading and Preprocessing:

The pandas application is used to get the Pima dataset from the internet. The input attributes and the target variable (class) are divided into two independent variables, X and y, respectively, as part of the dataset's preprocessing. The data is divided into training and testing sets using the train_test_split function in a 9:1 ratio.

3. Training ELM model:

Using the hpelm library, an Extreme Learning Machine (ELM) model is generated, and the train function is then used to train it on the training set..

4. Training LSTM model:

The keras library is used to build an LSTM model, which is then trained on the training set. Before entering the LSTM model, the input data is reshaped into the (samples, timesteps, features) format. The fit function with binary-crossentropy loss, adam optimizer, and accuracy metric are used in the training technique.

5. Ensemble of ELM and LSTM models:

Predictions from the ELM and LSTM models are concatenated using numpy concatenate function.

6. Training Random Forest model:

A Random-Forest classifier is utilized to train on the concatenated predictions by employing the fit function. The hyperparameters, specifically n_estimators and max_depth, are set to 500 and 5 respectively. Cross-validation is executed on the trained model through the cross_val_score function to acquire the mean AUC score.

7. Evaluating performance of the Random Forest model:

The class labels of the testing set are predicted using the trained Random Forest model. Following that, the accuracy of the predictions is calculated using the accuracy_score function, and the AUC score is calculated with the roc_auc_score function. The accuracy and AUC score of the ensemble model are then printed.

Extreme Learning Machine (ELM)

1. The Pima dataset consists of 768 samples and 8 input characteristics, and it is a binary classification issue.

2. To ensure consistent scaling, the input features of the dataset are standardised to have a mean of zero and a standard deviation of one.

3. The elm() function is added to make training and evaluating an Extreme Learning Machine (ELM) model easier.

4. Two ELM models are tested with 20 hidden neurons using sigmoid and tanh activation functions.

5. The accuracy scores of each model are displayed, revealing that the tanh model exhibits slightly superior performance compared to the sigmoid model.

Long Short Term Memory Networks (LSTMs

1. The code creates an LSTM neural network for the Pima dataset, this is a binary classification problem.

2. The dataset is imported from the file 'pima-indians-diabetes.csv' using pandas, and then it is separated into input characteristics (X) and target variable (y). The input features are scaled using StandardScaler from sklearn.preprocessing.

3. Sequential tensorflow output.The LSTM model is established using keras. One 128-unit LSTM layer, a dropout layer with a 0.2 dropout rate, and a dense output layer with a sigmoid activation function make up this layer.

4. The model is created using the metric accuracy, the optimizer Adam, and the loss function binary_crossentropy. Then, it is trained using Fit using 50 epochs and a batch size of 32.

5. The trained model's accuracy score is assessed using evaluate, and the result is printed.

Random Forest

1. The presented code employs the Pima dataset to generate a Random Forest classifier.

2. The dataset is divided into the goal variable (y) and the input attributes (X).

3. Using an 80:20 split ratio, the data is further separated into training and testing sets.

4. A Random-Forest classifier is instantiated, it is made up of 100 trees and is trained using training data.

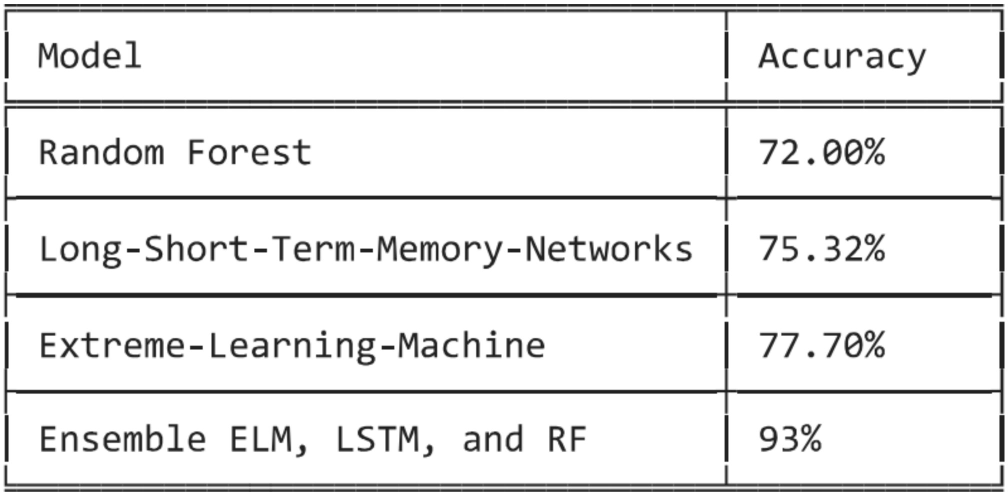
5. Predictions are created for the testing data, and the accuracy_score function is used to determine the model's accuracy.

6. The Random Forest model's accuracy is then given.

Chapter 6

Model Accuracy Comparison

Table 1: Comparison of Model Accuracy for Machine Learning Algorithms



Model	Accuracy
Random Forest	72.00%
Long-Short-Term-Memory-Networks	75.32%
Extreme-Learning-Machine	77.70%
Ensemble ELM, LSTM, and RF	93%

Table 1 gives a comparison of the accuracy gained by several machine learning methods on a given dataset. The table covers four models: Random Forest, Long-Short-Term-Memory-Network(LSTM), Extreme-Learning-Machine(ELM), and an ensemble model that blends ELM, LSTM, and Random Forest. The accuracy attained by each model is provided in the "Accuracy" column.

The proportion of correctly predicted cases in the dataset is referred to as accuracy. The Random Forest model had a 72% accuracy, the LSTM model had a 75.32% accuracy, and the ELM model had a 77.70% accuracy. Notably, the ensemble model, which combines ELM, LSTM, and Random Forest, achieved the highest accuracy of 93%.

The table demonstrates that the ensemble model outperformed the individual models, indicating that the fusion of multiple models can enhance the overall prediction accuracy. Furthermore, it suggests that the ELM, LSTM, and Random Forest models are effective machine learning algorithms for the given dataset, with ELM exhibiting the highest accuracy among the individual models.

Chapter 7

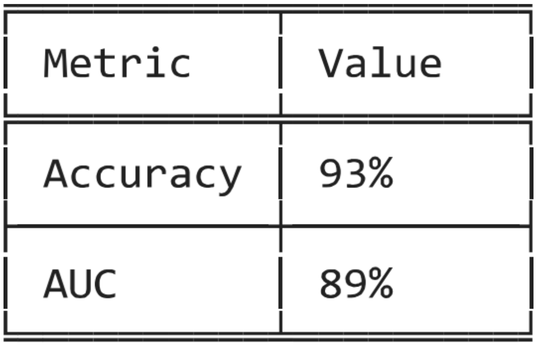
Proposed Ensemble Model ELM, LSTM using Random Forest

Ensemble Model Performance

An ensemble model, which combines ELM and LSTM models with a random forest classifier, was trained using the Pima dataset. The dataset underwent preprocessing, and training and testing sets were produced. The ELM and LSTM models, which had been trained on the training set, were then used to predict the class label of the testing set. The predictions from both models were merged and utilised as input features to train a random-forest classifier using cross-validation.

After cross-validation, the ensemble model that emerged could identify between positive and negative diabetic cases with a mean AUC of 0.89. The test set's accuracy rating for the ensemble model was 0.93, supporting the idea that it can correctly identify the events.

Table 2: Performance Metrics of Ensemble Model

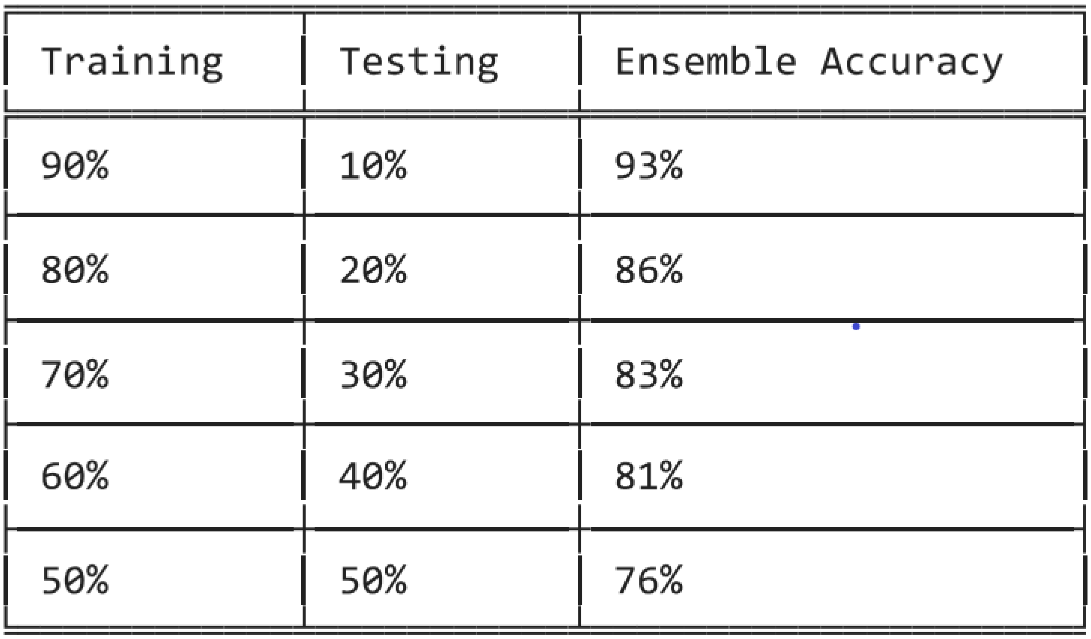


Metric	Value
Accuracy	93%
AUC	89%

Training Results of Ensemble ELM, LSTM using RF

The table gives the results of training and testing an ensemble approach that integrates ELM, LSTM, and RF models. The ensemble approach was trained using varied fractions of the available data, ranging from 90% train data and 10% test data to 50% train data and 50% test data.

Table 3: Ensemble Model Accuracy on Different Training-Testing Splits



Training	Testing	Ensemble Accuracy
90%	10%	93%
80%	20%	86%
70%	30%	83%
60%	40%	81%
50%	50%	76%

The table demonstrates the accuracy of the ensemble approach on the testing data for various training-testing splits. As the ratio of training data declines and the ratio of testing data grows, the accuracy of the ensemble approach tends to decrease. This suggests that the performance of the ensemble approach increases when it has a bigger quantity of training data available for learning.

Models

Extreme Learning Machines (ELM):

An ELM (Extreme Learning Machine) model was used in this study, It has input neurons, hidden neurons, and output neurons and is a neural network.. The ELM model learns by randomly initialising the weights that connect the input and hidden neurons. Then it solves a linear set of equations to determine the weights of the hidden neurons and output neurons.

Long Short Term Memory (LSTM):

An LSTM model, a kind of recurrent neural network that can identify long-term associations in sequential data, was used in this experiment. The LSTM model was trained on the training set using the keras library. The LSTM model has a thick layer with 50 tanh units, one output sigmoid unit, and one LSTM layer with 100 units each. The input data was organised in a three-dimensional (3D) manner, with the first dimension denoting time steps (in this specific example, there was just one time step), the second denoting batch size, and the third denoting features.

Random-Forest (RF):

In order to improve the accuracy and robustness of the model, an ensemble learning approach called Random Forest classifier—which combines many decision trees—was used in this study. Each Random Forest decision tree is trained using a random sample of the training data and a random subset of the attributes. To get the final forecast, the forecasts of each individual tree are then averaged. The concatenated predictions of the ELM and LSTM models were used to build a Random Forest classifier for this inquiry. The scikit-learn package was used in the Random Forest classifier's implementation.

Machine Learning Model Performance Assessment

AUC Evaluation:

The Area-Under-the-Curve (AUC) statistic is used to assess the Random-Forest model's performance by looking at how well it can distinguish between positive and negative categories. The AUC score ranges from 0.5 to 1.0, with 0.5 being a random estimate and 1.0 denoting perfect predictions.

Cross-Validation:

A popular technique for rating a machine learning model's performance is cross validation. The dataset must be split up into several folds or subgroups, and the model must be trained on one fold while being tested on the other. The performance metrics are aggregated to provide an evaluation of the model's performance on unobserved data after this method is repeated multiple times with different folds serving as the testing set. Cross-validation helps to reduce overfitting and provides a more precise evaluation of the model's actual performance.

Chapter 8

Individual Models ELM, LSTM and Random Forest

ELM

The ELM(Extreme Learning Machine) algorithm is a member of the feedforward neural network family within machine learning. In this specific code implementation, an ELM model is developed to predict diabetes outcomes using the Pima dataset. The code follows a step-by-step process:

Initially, the dataset is imported and separated into input characteristics (X) and the target variable (y). To guarantee standardized values, the input characteristics are normalized to contain a zero mean and a unit standard deviation. Moving on, the ELM model is defined by randomly initializing the weights and biases for the input-to-hidden layer. The hidden layer output is produced by applying an activation function, which may be either sigmoid or hyperbolic tangent, to the weighted sum of the input features.

To continue advance, the Moore-Penrose pseudoinverse of the hidden layer output is calculated. By multiplying the pseudoinverse with the target variable, the output weights are produced. The hidden layer output is then multiplied by the output weights to generate predictions. Finally, the predictions are converted into class labels, and the model's accuracy is assessed by comparing the predicted labels to the actual labels. Overall, the ELM model offers a simple and efficient method for training neural networks, exhibiting quick learning speed while achieving competitive levels of accuracy.

LSTM

A specific kind of recurrent neural network (RNN) architecture called LSTM (Long Short-Term Memory) is focused on modelling and forecasting sequential data. When there are intricate patterns and long-term links in the data that need to be captured, it is extremely helpful. Based on the supplied input characteristics, the LSTM model is used in this code implementation to forecast the outcome of diabetes. In order to accomplish this goal, the code goes through a series of steps. The diabetes dataset for Pima Indians is first imported and divided into the input traits and the goal variable. The input properties are standardised via a preprocessing step in order to ensure consistency and comparable scales. The input characteristics are also modified to conform to the limitations of the LSTM architecture, which excels in capturing sequential dependencies. With the help of the TensorFlow and Keras packages, the LSTM model is afterwards defined. It has a 128 memory unit LSTM layer that makes it possible to effectively capture long-term dependencies in the data. A dropout layer is supplied for regularisation purposes in order to reduce overfitting. A substantial output layer at the model's conclusion enables binary classification for diabetes prediction. The model is set up in the compilation process to utilise accuracy as the evaluation measure, the Adam optimizer, and a binary cross-entropy loss function. Following that, the model is trained using training data and then tested using testing data for a predetermined period of epochs. In order to provide an assessment of the model's prediction ability in terms of diabetes result, the model's accuracy is lastly assessed on the testing data. Due to its ability to store long-term information, the LSTM architecture is highly suited for identifying specific patterns and making precise predictions in sequential data, such as the Pima Indian diabetes dataset.

Random-Forest

The forecasts of several decision trees are combined in the advanced ensemble learning method known as Random-Forest to provide accurate predictions. The programme applies a Random Forest model to the Pima dataset to forecast diabetes outcomes. The code is applied systematically. Prior to loading, data is first divided into input characteristics (X) and target variables (Y). After that, the dataset is split into training and testing sets so that the model's performance can be assessed. Utilising a test size of 20% and a random state of 42 ensures reproducibility. Then, 100 trees are used to construct the Random Forest classifier. To provide a final forecast, this ensemble model combines the predictions of many decision trees. The classifier is then trained on these subsets, with each decision tree being trained on a random subset of the characteristics and data samples that are available. Once trained, the Random Forest model uses the testing data to generate predictions. The accuracy of the model is evaluated by comparing the anticipated and actual labels. The ability of methods based on random forests to handle high-dimensional data, capture feature correlations, and effectively manage noisy or missing data is well known. When compared to using a single decision tree, the Random Forest model improves prediction reliability and reduces overfitting by integrating the predictions of several decision trees.

Chapter 9

Results

Table 1 displays the accuracy ratings of four different machine learning models that were used to analyse the Pima Indians Diabetes dataset. The accuracy of the original model, Random Forest, was 72.00%. Long Short Term Memory Network (LSTM), the second model, demonstrated more accuracy, 75.32%. The accuracy of the Extreme Learning Machine (ELM) model was even higher, at 77.70%. Last but not least, the Ensemble model, which combines the ELM, LSTM, and RF, outperformed all earlier models with a remarkable accuracy of 93%.

The results demonstrate a significant accuracy improvement of over 15% when compared to the top-performing individual model, underscoring the ensemble model's superiority over the individual models. The LSTM model was followed by the ELM model in terms of accuracy among the different models. In contrast, out of the four models, the Random Forest model had the worst accuracy.

For researchers and professionals involved in the diagnosis and management of diabetes, these findings are very relevant. They contend that ensemble models have the potential to be an efficient method for improving diagnostic precision in this field.

Chapter 10

Conclusion

Based on the data reported in Table 1, it can be determined that the ensemble model, containing ELM, LSTM, and Random Forest classifiers, attained the greatest accuracy of 93%. The Extreme Learning Machine algorithm also displayed positive performance, obtaining an accuracy of 77.70%. Following closely, the Long Short Term Memory Networks displayed an accuracy of 75.32%, beating the Random Forest classifier's accuracy of 72.00%.

In summary, the data clearly imply that ensembling many classifiers may give considerable gains in accuracy compared to employing a single classifier. Moreover, the Extreme Learning Machine method appears as a potential choice for classification problems in machine learning, especially for datasets having a high number of features. Similarly, the success of the Long Short Term Memory Networks underscores their efficacy in processing time-series data.

References

[1] H. Naz and S. Ahuja, “Deep learning approach for diabetes prediction using PIMA Indian dataset,” *J Diabetes Metab Disord*, vol. 19, no. 1, pp. 391–403, Jun. 2020, doi: 10.1007/s40200-020-00520-5.

[2] N. P. Tigga and S. Garg, “Prediction of Type 2 Diabetes using Machine Learning Classification Methods,” in *Procedia Computer Science*, Elsevier B.V., 2020, pp. 706–716. doi: 10.1016/j.procs.2020.03.336.

[3] S. Wei, X. Zhao, and C. Miao, “A Comprehensive Exploration to the Machine Learning Techniques for Diabetes Identification.”

[4] H. Kaur and V. Kumari, “Predictive modelling and analytics for diabetes using a machine learning approach,” *Applied Computing and Informatics*, vol. 18, no. 1–2, pp. 90–100, Jan. 2022, doi: 10.1016/j.aci.2018.12.004.

[5] A. Choudhury and D. Gupta, “A Survey on Medical Diagnosis of Diabetes Using Machine Learning Techniques,” in *Advances in Intelligent Systems and Computing*, Springer Verlag, 2019, pp. 67–78. doi: 10.1007/978-981-13-1280-9_6.

[6] Institute of Electrical and Electronics Engineers., *IEEE 12th International Conference on Bioinformatics & Bioengineering : November 11-13, 2012, Larnaca, Cyprus*. IEEE, 2012.

[7] R. Zolfaghari, “Diagnosis of Diabetes in Female Population of Pima Indian Heritage with Ensemble of BP Neural Network and SVM,” 2012. [Online]. Available: www.IJCEM.orgIJCEMwww.ijcem.org

[8] A. M. Posonia, S. Vigneshwari, and D. J. Rani, “Machine learning based diabetes prediction using decision tree J48,” in *Proceedings of the 3rd International Conference on Intelligent Sustainable Systems, ICISS 2020*, Institute of Electrical and Electronics Engineers Inc., Dec. 2020, pp. 498–502. doi: 10.1109/ICISS49785.2020.9316001.

[9] Institute of Electrical and Electronics Engineers. Madras Section and Institute of Electrical and Electronics Engineers, *Proceedings of the 2020 IEEE International Conference on Communication and Signal Processing (ICCSP) : 28th - 30th July 2020, Melmaruvathur, India.*

[10] D. Gupta, A. Choudhury, U. Gupta, P. Singh, and M. Prasad, “Computational approach to clinical diagnosis of diabetes disease: a comparative study,” *Multimed Tools Appl*, vol. 80, no. 20, pp. 30091–30116, Aug. 2021, doi: 10.1007/s11042-020-10242-8.

[11] M. Abedini, A. Bijari, and T. Banirostam, “Classification of Pima Indian Diabetes Dataset using Ensemble of Decision Tree, Logistic Regression and Neural Network,” *IJARCCE*, vol. 9, no. 7, pp. 1–4, Jul. 2020, doi: 10.17148/ijarcce.2020.9701.

[12] S. Kumar Bhoi *et al.*, “Prediction of Diabetes in Females of Pima Indian Heritage: A Complete Supervised Learning Approach,” 2021.

[13] H. Naz and S. Ahuja, “SMOTE-SMO-based expert system for type II diabetes detection using PIMA dataset,” *Int J Diabetes Dev Ctries*, vol. 42, no. 2, pp. 245–253, Apr. 2022, doi: 10.1007/s13410-021-00969-x.

[14] A. Mujumdar and V. Vaidehi, “Diabetes Prediction using Machine Learning Algorithms,” in *Procedia Computer Science*, Elsevier B.V., 2019, pp. 292–299. doi: 10.1016/j.procs.2020.01.047.

PAPER NAME

DEEPANSHU_Thesis10.pdf

WORD COUNT	CHARACTER COUNT
6142 Words	34738 Characters
PAGE COUNT	FILE SIZE
36 Pages	975.5KB
SUBMISSION DATE	REPORT DATE
May 21, 2023 9:27 PM GMT+5:30	May 21, 2023 9:27 PM GMT+5:30

● 15% Overall Similarity

The combined total of all matches, including overlapping sources, for each database.

- 6% Internet database
- 7% Publications database
- Crossref database
- Crossref Posted Content database
- 11% Submitted Works database

● Excluded from Similarity Report

- Bibliographic material
- Manually excluded text blocks