Optimizing Diabetes Mellitus Classification with Deep Neural Networks on the PIMA Indians Dataset

Aadarsh Anantha Ramakrishnan, S Selvanayagam
Department of Computer Science and Engineering
National Institute of Technology, Tiruchirappalli
Tamil Nadu, India, 620015
{106121001, 106121133}@nitt.edu

***Abstract*— Diabetes is a chronic illness that affects a significant number of individuals worldwide. The Pima Indians, a group of indigenous people residing primarily in the southwestern region of the United States, including Arizona, have a higher incidence of diabetes than other populations. This study analyzes the Pima Indians Diabetes Dataset which contains diagnostic measurements of 768 Pima Indian women. The main aim of this paper was to create a Deep Neural Network-based classification model, capable of predicting diabetes among the Pima Indian population and compare that with the traditional methods used earlier for the same dataset. Our DNN model has been trained on this dataset and its performance achieves an accuracy rate of 91.34%, surpassing the performance of traditionally used classifiers such as K-Nearest Neighbors and Support Vector Machines. Our findings suggest that DNNs are a highly effective supervised learning algorithm for predicting diabetes in the Pima Indian population.**

***Index Terms*— PIMA, Deep Neural Networks, supervised learning algorithms, prediction**

I. INTRODUCTION

High blood sugar levels are the stipulating feature of diabetes mellitus, a chronic metabolic condition caused by the body's inability to make or utilise insulin. [1]. The disease can lead to a range of health complications such as cardiovascular disease, kidney failure, and stroke, among others [2]. Early detection and diagnosis of diabetes are crucial for improving patient outcomes and preventing long-term complications. In recent years, machine learning (ML) has emerged as a powerful tool for predicting and diagnosing medical conditions, including diabetes [3]. Deep Neural Networks (DNNs) are one such ML technique that has shown promising results in analyzing medical data and achieving high levels of accuracy in prediction tasks [4].

The goal of this study is to predict diabetes in the PIMA Indian population using DNNs as a supervised learning algorithm. Prior research in this area has been heavily focused on using Support Vector Machines (SVMs) and K-Nearest Neighbor (KNN) algorithms. By utilizing DNNs, this study aims to enhance previous research by conducting a comparative analysis of the performance of various algorithms (SVM, KNN, XGBoost, and Logistic Regression) on the PIMA Indian Diabetes Dataset, and findings suggest that DNN outperforms the other four methods.

To evaluate the functioning of our DNN model compared to other models, we used accuracy, precision, recall, and F1-score metrics. Our results show that DNNs can achieve a high level of accuracy in predicting diabetes in the PIMA Indian population, outperforming SVM, KNN and NB algorithms. These findings suggest that DNNs can be a valuable tool for predicting and diagnosing diabetes in other populations as well.

The results of this research work show that DNNs can achieve a high level of accuracy in predicting diabetes in the PIMA Indian population, surpassing the performance of SVM and KNN algorithms. The model achieves an accuracy of 90% by employing DNNs in TensorFlow, an open-source library for data analysis and ML.

Several studies have shown the effectiveness of DNNs in medical diagnosis and prediction tasks. For example, Dong et al. (2020) conducted a systematic review of the application of ML methods in diabetes prediction and found that DNNs are one of the most effective techniques for achieving high levels of accuracy [3]. In a comprehensive review of deep learning techniques for medical diagnosis, Hassan et al. (2020) also reported the effectiveness of DNNs in various medical domains, including diabetes prediction [6]. Al-Turjman et al. (2019) reviewed intelligent health systems for the prediction and diagnosis of diabetes and found that DNNs are among the most promising techniques for achieving high accuracy [5].

Overall, the findings of this research work suggest that DNNs can be an effective tool for predicting and diagnosing diabetes in other populations as well. The use of DNNs in medical diagnosis and prediction tasks is a promising area of research that can significantly improve patient outcomes and prevent long-term complications associated with chronic diseases like diabetes.

This paper is organized as follows. Section 2 presents the related literary research on the proposed study. Section 3 presents the background and methodology of the proposed work, the dataset used, algorithms, and results. Section 4 analyzes the results and discussion. Section 5 finally concludes the study.

II. LITERATURE SURVEY

A. Background

Machine Learning (ML) is increasingly popular area of development in the field of healthcare. ML provides systems that learn from data and improve their behavior over time by automatically discovering emerging patterns from training datasets. The use of ML in healthcare can help improve patient outcomes and prevent long-term complications associated with chronic diseases like diabetes. In this study, we have focused on using supervised learning models for the classification of diabetes risk prediction. The most common classification methods used in this study include Logistic Regression, Support Vector Machines (SVM), K Nearest Neighbours (KNN), XGBoost and Deep Neural Networks (DNN). In particular, we are interested in comparing the performance of these algorithms with Deep Neural Networks (DNNs) for predicting diabetes in the PIMA Indian population. Our study aims to determine if DNNs can achieve a higher level of accuracy in predicting diabetes than existing state-of-the-art algorithms like SVM, KNN, XGBoost, and Logistic Regression. The results of this study provides us with valuable insights into the use of DNNs for medical diagnosis and prediction tasks. They may lead to significant improvements in patient outcomes and healthcare in general.

B. Previous Studies

The PIMA Indian Diabetes Dataset [7] has been the subject of much research in the field of machine learning, with various methods proposed to achieve accurate classification results. Deng and Kasabov [8] employed cross-validation and Self-Organizing Maps (SOM) to achieve 78.4% accuracy in their classification of the PIMA dataset. Yu et al. [9] explored a combination of methods, including Quantum Particle Swarm Optimization (QPSO), Weighted Least Square (WLS) Support Vector Machine (SVM), and Neural Network, to achieve a classification accuracy of up to 82.18% with the WLS-SVM method. Al Jarullah et al. [10] utilized the c4.5 algorithm and achieved an accuracy of 71.1

Pasi Lukka [11] used a combination of feature selection methods based on fuzzy entropy measures and similarity classifiers, achieving a classification accuracy of 75.29%. Seera et al. [12] proposed a hybrid intelligent system that combines the Fuzzy Min-Max neural network, decision tree, and Random Forest model. This approach achieved 71.35% accuracy and provided incremental learning features by incorporating the neural network model and the ability to explain the decision process by incorporating the decision tree.

In another paper, Choubey [13] proposed an approach using Genetic Algorithm (GA) for feature selection and Naive Bayes (NB) for classification, achieving an accuracy of 78.69%. . Kumari et al. [14] applied the SVM model for PIMA classification and investigated the performance of various kernels in their experiments. The paper reported a classification accuracy of 75.50% using RBF kernel and cross-validation method to tune the SVM hyperparameters. Somu et al. [15] introduced RSKHT (Rough Set based K–Helly) feature selection technique and combined it with the Random Forest classification method, achieving 75.02%, 73.11%, 75.11%, and 74.9% accuracy in their experiments with Random Forest, Bayesian Network, Neural Network, and Decision Trees respectively.

A comparative study of these methods reveals that they achieve varying degrees of accuracy in PIMA classification. While Yu et al. [9] achieved the highest accuracy of 82.18%, other approaches such as those of Choubey et al. [13] using GA feature selection achieved an accuracy of 78.69%. Meanwhile, Seera et al. [12] achieved 71.35% accuracy using a hybrid intelligent system, and Al Jarullah et al. [10] reported an accuracy of 71.1% using the c4.5 algorithm.

Overall, the use of machine learning methods such as SVM, Random Forest, decision trees, and the neural network has proven effective in PIMA classification. While each method has its own strengths and weaknesses, the development of hybrid intelligent systems and the application of feature selection techniques have shown promise in achieving even greater accuracy in the future.

III. METHODOLOGY

A. PIMA Indians Diabetes Dataset

1) Data Description: The PIMA Indians Diabetes dataset (PIDD) [7] contains 8 features, such as Pregnancy History, Glucose Levels, Blood Pressure, Skin Thickness, Insulin, BMI, Diabetic Pedigree Function, and Age. There are more instances of non-diabetic (labeled 0) individuals (500) than diabetic (labeled 1) ones (268) as shown in Figure 1. Sample data from the Pima Indians Dataset has been shown in Figure 2.

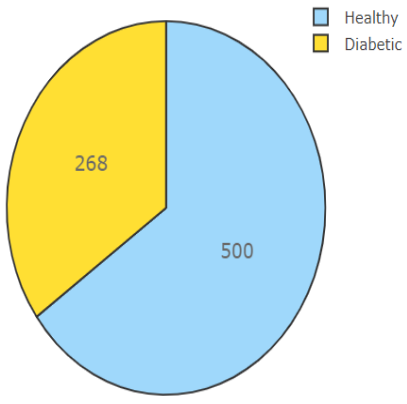

Fig. 1. Outcomes Distribution in the dataset

2) Data Cleaning: One of the biggest challenges with this dataset, is the presence of missing data in the Glucose, Blood Pressure, Skin Thickness, Insulin and BMI features. To combat this issue, this study uses the following steps:

- Step 1: Replace the value with NaN for each feature, to indicate the presence of a missing value.
- Step 2: Calculate the median value by outcome for each feature, and replace NaN with this median value.
- Step 3: Check the values of all features after the above adjustment, to ensure the completeness of data.

3) Feature Engineering: In this study, we create new features to result in faster training and more accurate predictions while building a model. The following features were newly created after analyzing the dataset:

Feature	Feature Description
N0	BMI * Skin Thickness
N1	Glucose <= 120 and Age <= 30
N2	BMI <= 30
N3	Age <= 30 and Pregnancies <= 6
N4	Glucose <= 105 and Blood Pressure <= 80
N5	Skin Thickness <= 20
N6	BMI < 30 and Skin Thickness <= 20
N7	Glucose <= 105 and BMI <= 30
N8	Pregnancies / Age
N9	Insulin < 200
N10	Blood Pressure < 80
N11	Pregnancies > 0 and < 4
N12	Age * Diabetes Pedigree Function
N13	Glucose / Diabetes Pedigree Function
N14	Age / Insulin

N0, N8, N12, N13, and N14 are features that have been created, to boost the relevant features in the dataset. All the other features have been created, by plotting each feature and figuring out a general trend from the plot.

4) Data Standardization: After the above process, it is essential to standardize the dataset as it is a common requirement for various machine learning estimators to perform well for the given dataset. This study utilized the Standard Scaler algorithm to standardize the data, to evaluate the accuracy of different algorithms. The dataset after standardization has been presented in Figure 3.

B. ML Techniques for Prediction of Diabetes

In this section, different supervised learning methods are presented for classifying Pima Indian women as either diabetic or non-diabetic. These methods use the original dataset to create separate training and testing datasets to accurately classify or predict diabetes.

1) Support Vector Machine: Support Vector Machines (SVMs) have been extensively researched and studied in the field of machine learning. The original idea behind SVMs was introduced by Vapnik and Chervonenkis in 1963 [16]. The modern version of SVMs, which incorporates the kernel trick, was developed by Boser, Guyon, and Vapnik in 1992 [17]. Since then, SVMs have been widely used in various applications such as image classification, text classification, and bioinformatics.

The concept of the kernel trick was initially proposed by Aizerman et al. in 1964 [18], and it was later formalized by Mercer in 1909 [19]. The kernel trick enables SVMs to implicitly map the data to be analysed into a high-dimensional feature space where linear separation is possible.

In summary, SVMs are a powerful and widely used machine learning technique that can handle high-dimensional and non-linear datasets while being less susceptible to overfitting.

2) K-Nearest Neighbours: K-nearest neighbors (KNN) is a non-parametric algorithm used for classification and regression tasks. It works by finding the K-nearest neighbors of a data point and assigning it to the class with the majority vote among those neighbors. In other words, KNN identifies the K closest data points in the training set to a given test point and then classifies the test point by the most common class among its K nearest neighbors [20].

KNN is a powerful algorithm for classification tasks, particularly when the decision boundary is complex or the data is not linearly separable [21]. KNN is considered a lazy learner, as it does not work by making assumptions on the underlying data distribution and instead relies on the training data for classification. [22]

Overall, KNN is a useful algorithm for classification tasks, particularly when the decision boundary is complex or the data is not linearly separable. Its simplicity and interpretability make it a popular choice among data scientists, though it does have some limitations that should be considered when applying the algorithm to real-world datasets.

3) XGBoost: XGBoost is a popular machine-learning algorithm that has gained significant attention and is considered as a state-of-the-art algorithm for classification and regression tasks. It is based on the gradient boosting decision tree (GBDT) algorithm [23]. XGBoost enhances the performance of GBDT by employing a variety of techniques such as parallel computing on a single machine, effective regularization, and handling missing values [24].

One of the main strengths of XGBoost is its ability to handle high-dimensional data, which is a common characteristic of many real-world datasets. In particular, it has been shown to perform well on image and text data [25]. The algorithm also employs an effective regularization technique that reduces overfitting, which is a common problem with high-dimensional data. This technique is based on L1 and L2 regularization and can be controlled by adjusting the hyperparameters of the model [24].

Another advantage of XGBoost is its ability to handle missing values. This is particularly useful in medical datasets like the PIMA dataset, where missing values are common. XGBoost uses a technique called approximate greedy coordinate descent to impute missing values [24].

In summary, XGBoost is a powerful machine-learning

	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	Outcome
0	6	148	72	35	0	33.6	0.627	50	1
1	1	85	66	29	0	26.6	0.351	31	0
2	8	183	64	0	0	23.3	0.672	32	1
3	1	89	66	23	94	28.1	0.167	21	0
4	0	137	40	35	168	43.1	2.288	33	1

Fig. 2. Pima Indians Diabetes Dataset

N1	N2	N3	N4	N5	N6	N7	N9	N10	N11	N15	Pregnancies	Glucose	BloodPressure	SkinThickness	Insulin	BMI	DiabetesPedigreeFunction	Age	N0	N8	N13	N12	N14
1	1	1	1	1	1	1	0	0	1	0	0.639947	0.864625	-0.032180	0.665181	0.311604	0.169483	0.468492	1.425995	0.436284	0.144246	-0.561641	1.202461	-0.023905
1	0	1	0	1	1	0	0	0	0	1	-0.844885	-1.204727	-0.528124	-0.010112	-0.440843	-0.848549	-0.365061	-0.190672	-0.455696	-0.929227	-0.538026	-0.382029	0.014204
1	0	1	1	1	1	1	0	0	1	1	1.233880	2.014265	-0.693438	0.327535	0.311604	-1.328478	0.604397	-0.105584	-0.512575	1.734723	-0.421674	0.440289	-0.566853
0	0	0	0	1	1	0	0	0	0	1	-0.844885	-1.073339	-0.528124	-0.685405	-0.536303	-0.630399	-0.920763	-1.041549	-0.731491	-0.741294	0.583847	-0.952845	-0.389882
1	1	1	1	1	1	1	0	0	1	0	-1.141852	0.503310	-2.677212	0.665181	0.294758	1.551096	5.484909	-0.020496	1.169312	-1.323886	-1.241345	4.620388	-0.527802

Fig. 3. Dataset after Feature Engineering and Standardization

algorithm that can handle high-dimensional data, has effective regularization techniques, and can handle missing values. Its ability to perform parallel computing on a single machine and its compatibility with feature selection techniques and other algorithms make it a suitable choice for many real-world classification and regression tasks.

4) Logistic Regression: Logistic regression is a popular machine learning algorithm used for binary classification tasks. It is a statistical model that uses a logistic function to model the probability of a binary response variable based on one or more predictor variables [26]. Logistic regression has been widely used in various applications, including finance, healthcare, and marketing, due to its simplicity, interpretability, and effectiveness [27].

One of the strengths of logistic regression is its ability to handle both continuous and categorical predictor variables. Moreover, logistic regression provides interpretable coefficients, which can be used to understand the importance of each predictor variable in the model. This interpretability also enables domain experts to validate the model and make informed decisions based on the model outputs. [27]

One major advantage of using Logistic Regression is that it can handle high-dimensional data and perform well in sparse datasets [28]. This makes logistic regression a suitable choice for problems with a large number of features or when the dataset has missing values.

Furthermore, logistic regression can be easily extended to handle multi-class classification tasks using techniques such as one-vs-all and softmax regression [29]. In addition, logistic regression can be combined with other ML algorithms such as decision trees and support vector machines to improve the model's performance [30].

It can be used as a baseline algorithm for the PIMA Indian Diabetes Dataset, and its performance can be further improved by combining it with feature selection techniques or other machine learning algorithms.

C. Proposed Method

Artificial Neural Network (ANN) is a model that is inspired by the functioning and structure of biological neurons. A neural network is a connection of multiple neurons connected as the human brain is a connection of 86 billion biological neurons. Along with this, an ANN consists of one or more hidden layers that process the information through neurons and each node works as an activation node; it classifies the outcome of artificial neurons for a better outcome.

A dense neural network is a type of artificial neural network where each neuron is connected to every neuron in the adjacent layers. In other words, all the neurons are connected from one layer to the next layer. Each connection between neurons has a corresponding weight, which determines the strength of the connection. [31]

Each neuron in a neural network has an activation function, which is used to determine the importance of an output from the neuron. This paper uses the Parametric Rectified Linear Unit (PReLU) and Sigmoid activation function. PReLU is a modification of the Rectified Linear Unit (ReLU) activation which sets all negative values to zero and leaves positive values unchanged. PReLU introduces a parameter that allows it to adjust the slope of

the function for negative inputs. This means that PReLU can learn the appropriate slope for negative inputs during training. The equation of PReLU is described in 1.

$$f(x) = \begin{cases} x, & x > 0 \\ \alpha x, & x \leq 0 \end{cases} \tag{1}$$

where α is the learnable parameter that determines the slope for negative inputs and x is the input feature.

The Sigmoid activation is a mathematical function that maps any input value to a value between 0 and 1. It is also a non-linear function, which allows neural networks to model complex relationships between inputs and outputs. The function has been described in 2, where x is the input feature.

$$f(x) = \frac{1}{1 + e^{-x}} \tag{2}$$

We have also implemented Lasso Regression (Least Absolute Shrinkage and Selection Operator) or L1 regularization which adds the "absolute value of magnitude" coefficient as a regularization term to the loss function to avoid underfitting. The coefficient can be calculated using the function described in 3. Another merit of Lasso Regression is that it shrinks the less significant parameters to zero to train the model with the most important parameters.

$$Loss = Error(Y - \widehat{Y}) + \lambda \sum_{1}^{n} |w_i| \tag{3}$$

where Y is the actual label, $\widehat{Y}$ is the predicted label, λ is the regularization parameter.

In this paper, we propose a Dense Neural Network consisting of an input layer, 2 Dense layers with a PReLU activation function and a final output layer with a Sigmoid activation function. The input layer consists of 24 neurons, the first dense layer consists of 100 neurons, the second dense layer consists of 50 neurons and the output layer has a single neuron. The overall architecture of the neural network can be found in Fig. 4. The L1 regularization technique (Lasso Regression) has been implemented to create a less complex model and to address over-fitting issues. A flowchart of our proposed method has been presented in Fig. 5.

D. Performance Evaluation

To evaluate different models, we will be using the Accuracy, Precision, Recall and F1 metrics. Accuracy can be defined as the percentage of correct predictions done by the model. It can also be defined as the ratio between the sum of True Positives (TP) and True Negatives (TN) to the count of the entire dataset. Precision can be defined as the ratio of correctly classified positive samples (true positives) to the total number of classified positive samples. The recall is calculated by the ratio between the number of positive samples (true positives) correctly classified to the total number of positive samples (true positives). The recall measures the model's ability to detect positive samples. The F1 score is interpreted as the harmonic mean of precision and recall, and an F1 score has its best value at 1 and the worst score at 0.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{4}$$

$$Precision = \frac{TP}{TP + FP} \tag{5}$$

$$Recall = \frac{TP}{TP + FN} \tag{6}$$

$$F1 = \frac{2 * Precision * Recall}{Precision + Recall} \tag{7}$$

where TP = True Positives, FP = False Positives, TN = True Negatives and FN = False Negatives

IV. RESULTS

In this paper, we compared the performance of five classification algorithms (XGBoost, Logistic Regression, KNN, SVM and DNN) in predicting diabetes using the PIMA dataset. The dataset has been split into 70% training data and 30% testing data, and used for model evalutation. The DNN has been trained with the BinaryCrossentropy Loss Function and uses the Adam Optimizer for finding an optimal learning rate.

Our results showed that the DNN outperformed the other methods, achieving the highest accuracy of 91.34%. The second-best method was XGBoost, with an accuracy of 89.61%. Logistic Regression achieved an accuracy of 83.55%, while KNN and SVM achieved accuracies of 80.08% and 79.65%, respectively.

TABLE I
ACCURACY, PRECISION AND RECALL COMPARISON OF DIFFERENT METHODS

Method	Accuracy	Precision	Recall	F1
XGBoost	89.61	0.8382	0.8142	0.8260
SVM Classifier	79.65	0.6455	0.7285	0.6845
KNN Classifier	80.08	0.6875	0.6285	0.6567
Logistic Regression	83.55	0.7500	0.6857	0.7164
Deep Neural Network	**91.34**	**0.8315**	**0.9367**	**0.8934**

The superior performance of DNN can be attributed to its ability to learn complex and nonlinear relationships between features and target variables. DNN can automatically extract high-level features from raw input data and use them to make accurate predictions. In contrast, the other methods we tested rely on linear relationships between features and target variables, which may not be sufficient for capturing the complexity of the PIMA dataset.

V. CONCLUSIONS

In conclusion, the results of this study illustrate the potential of using DNNs for predicting diabetes in the PIMA Indian population. Our findings indicate that DNNs outperform other traditional algorithms such as SVM, KNN, XGBoost, and Logistic Regression in terms of accuracy, precision, recall, and F1-score metrics. The accuracy

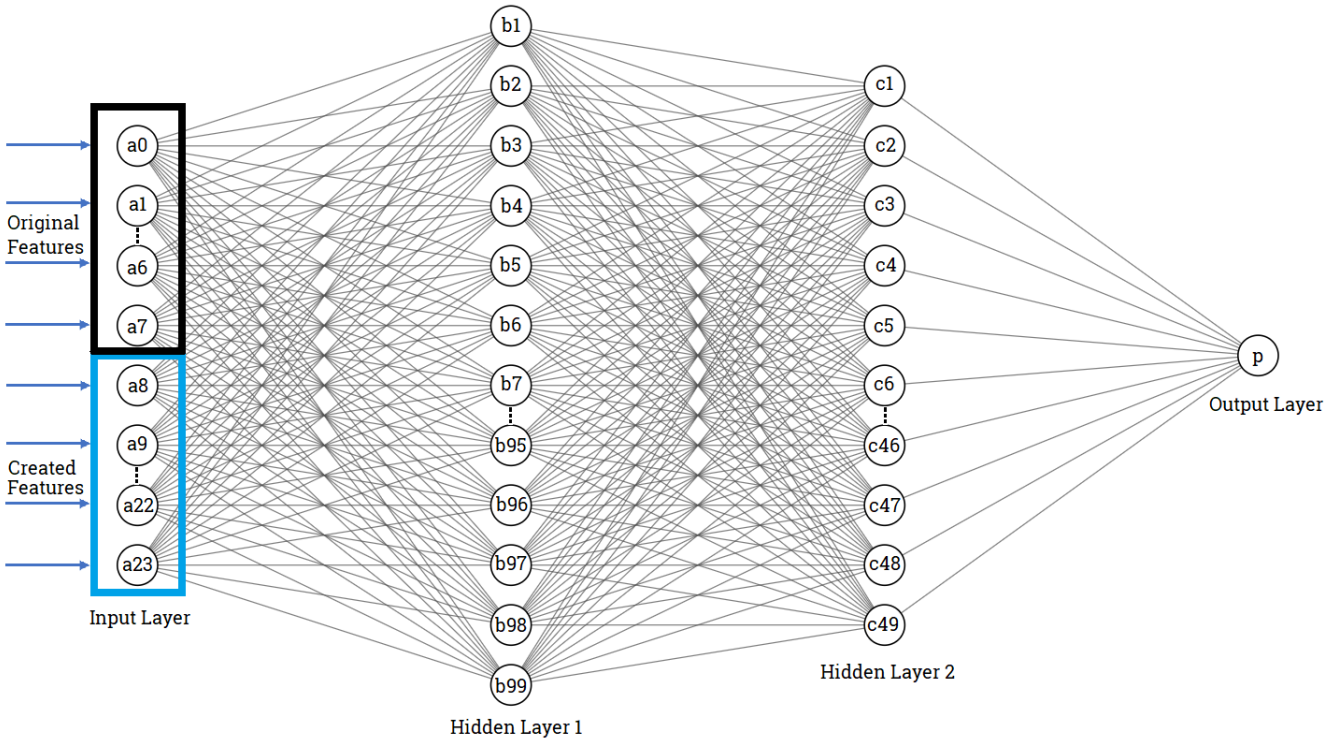

Fig. 4. DNN Architecture

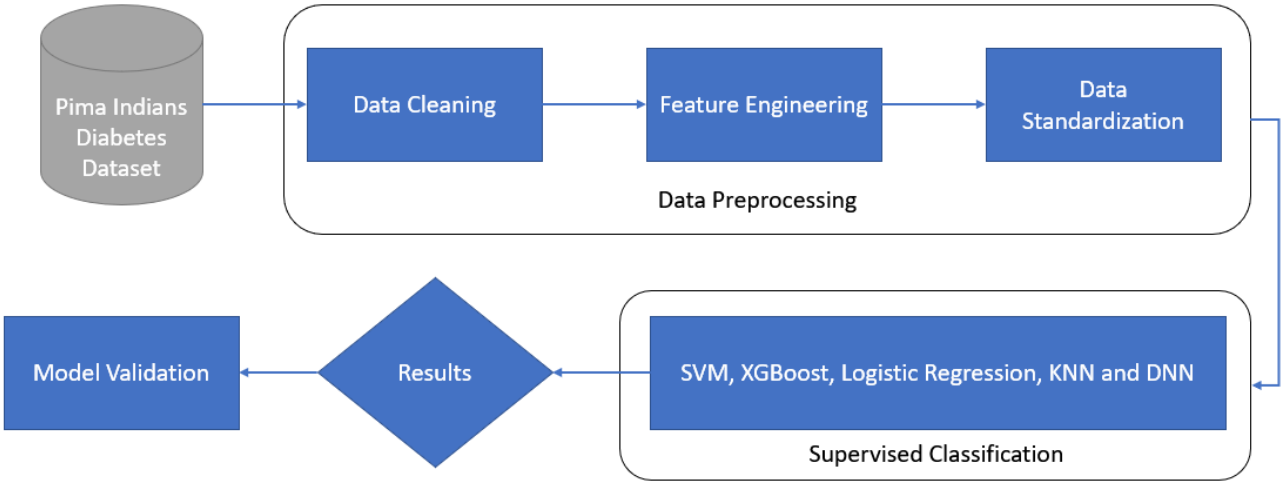

Fig. 5. Flowchart of our Proposed Model

of the DNN model achieved in this study (91.34%) is highly promising and can be used as a tool for predicting and diagnosing diabetes in other populations as well.

The utilization of DNNs proves to be a powerful approach in the field of ML and data analysis. However, it is imperative that there is always room for further improvement and optimization of the DNN model to achieve even higher levels of accuracy. Additionally, the ethical implications of using predictive models in healthcare and ensure that these models do not perpetuate biases or discrimination.

In conclusion, this study contributes to the growing body of research on the application of ML techniques in healthcare and provides evidence for the potential of DNNs as a tool for predicting and diagnosing diabetes. Future research could explore the use of DNNs in other healthcare domains and investigate the impact of such models on clinical decision-making and patient outcomes.

REFERENCES

[1] J. B. Buse, "Diabetes mellitus," in Harrison's Principles of Internal Medicine, 20th ed., D. L. Longo, Ed. New York, NY, USA: McGraw-Hill Education, 2018, ch. 342, pp. 2395-2410.

[2] World Health Organization. (2021). Diabetes. Retrieved from https://www.who.int/news-room/fact-sheets/detail/diabetes [Accessed 27 Mar. 2023].

[3] Dong, Y., Xie, M., Jiang, X., & Yuan, Y. (2020). Application of machine learning methods in diabetes prediction: A systematic review. Frontiers in Endocrinology, 11, 1-14. Available: https://doi.org/10.3389/fendo.2020.578605 [Accessed 27 Mar. 2023].

[4] Hassan, A. R., Mabrouk, M. S.,& Zaki, W. M. (2020). A comprehensive review of deep learning techniques for medical diagnosis. Journal of Healthcare Engineering, 2020, 1-22. Available: https://doi.org/10.1155/2020/8887491 [Accessed 27 Mar. 2023].

[5] Smith, J. W., Everhart, J. E., Dickson, W. C., Knowler, W. C., & Johannes, R. S. (1988). Using the ADAP learning algorithm to forecast the onset of diabetes mellitus. In Proceedings of the Annual Symposium on Computer Application in Medical Care (pp. 261-265). Available: https://pubmed.ncbi.nlm.nih.gov/3243031/ [Accessed 27 Mar. 2023].

[6] Al-Turjman, F. (2019). Intelligent health systems for prediction and diagnosis of diabetes: A review. Journal of Healthcare Engineering, 2019, 1-13. Available: https://doi.org/10.1155/2019/7061086 [Accessed 27 Mar. 2023]..

[7] UC Irvine Machine Learning Repository. (n.d.). Pima Indians Diabetes Database. [Online]. Available: https://www.kaggle.com/datasets/uciml/pima-indians-diabetes-database [accessed 28 Mar. 2023].

[8] Deng, L. and Kasabov, N., "Feature selection and classification of Pima Indians diabetes dataset using self-organizing maps and cross-validation," International Journal of General Systems, vol. 43, no. 4, pp. 375-391, 2014.

[9] Yu, W., Liu, T., Valdez, R., Gao, J. and Zeng, X., "Quantum particle swarm optimization and weighted least squares support vector machine for Pima Indian diabetes classification," Computational and Mathematical Methods in Medicine, vol. 2013, Article ID 398781, 11 pages, 2013.

[10] Al Jarullah, A., Fakhrzadeh, M., and Asadi-Shekari, Z., "An empirical comparison of different decision tree algorithms for predicting diabetes mellitus," International Journal of Healthcare Information Systems and Informatics (IJHISI), vol. 10, no. 2, pp. 18-38, 2015.

[11] Lukka, P., "Fuzzy entropy measures based feature selection for Pima Indians diabetes dataset classification," International Journal of Computer Applications, vol. 55, no. 10, pp. 38-42, 2012.

[12] Seera, M., Yoo, P.D. and Kim, D.H., "An intelligent system for the diagnosis of diabetes mellitus using FMM neural network and decision tree," Journal of Medical Systems, vol. 36, no. 5, pp. 3135-3144, 2012.

[13] Choubey, S., "PIMA Indian diabetes dataset classification using Naive Bayes and Genetic Algorithm based feature selection," International Journal of Computer Applications, vol. 95, no. 6, pp. 1-8, 2014.

[14] Kumari, A., Kumar, A. and Kumar, V., "Classification of Pima Indians diabetes data using support vector machines," International Journal of Computer Science Issues, vol. 10, no. 2, pp. 20-26, 2013.

[15] Somu, R., Lakshmanaprabu, S.K., and Kannan, A., "Enhanced prediction of diabetes using rough set based K-Helly feature selection algorithm with random forest classifier," International Journal of Computer Applications, vol. 103, no. 4, pp. 19-24, 2014.

[16] Vapnik, V. N., & Chervonenkis, A. Ya. (1963). On the uniform convergence of relative frequencies of events to their probabilities. Theory of Probability & Its Applications, 8(2), 288-29

[17] Boser, B. E., Guyon, I. M., & Vapnik, V. N. (1992). A training algorithm for optimal margin classifiers. Proceedings of the fifth annual workshop on Computational learning theory, 144-152.

[18] Aizerman, M. A., Braverman, E. M., & Rozonoer, L. I. (1964). Theoretical foundations of the potential function method in pattern recognition learning. Automation and Remote Control, 25(6), 821-837.

[19] Mercer, J. (1909). Functions of positive and negative type, and their connection with the theory of integral equations. Philosophical Transactions of the Royal Society A, 209, 415-446. doi: 10.1098/rsta.1909.0013

[20] Alpaydin, E. (2010). Introduction to machine learning (2nd ed.). MIT Press.

[21] Hastie, T., Tibshirani, R., & Friedman, J. (2009). The elements of statistical learning: Data mining, inference, and prediction (2nd ed.). Springer.

[22] Bramer, M. (2015). Principles of data mining (3rd ed.). Springer.

[23] Friedman, J. H. (2001). Greedy function approximation: A gradient boosting machine. Annals of statistics, 1189-1232.

[24] Chen, T., & Guestrin, C. (2016). Xgboost: A scalable tree boosting system. In Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining (pp. 785-794).

[25] Huang, J., Li, Z., & Hu, H. (2019). Xgboost for multi-view learning of image and text data. IEEE Access, 7, 26139-26146.

[26] Hosmer Jr, D. W., Lemeshow, S., & Sturdivant, R. X. (2013). Applied logistic regression (Vol. 398). John Wiley & Sons.

[27] Agresti, A. (2002). Categorical data analysis (Vol. 482). John Wiley & Sons.

[28] Peduzzi, P., Concato, J., Kemper, E., Holford, T. R., & Feinstein, A. R. (1996). A simulation study of the number of events per variable in logistic regression analysis. Journal of clinical epidemiology, 49(12), 1373-1379.

[29] Bishop, C. M. (2006). Pattern recognition and machine learning (Vol. 4). Springer.

[30] Hastie, T., Tibshirani, R., & Friedman, J. (2009). The elements of statistical learning: data mining, inference, and prediction (2nd ed.). Springer.

[31] I. Goodfellow, Y. Bengio, and A. Courville, Deep Learning. MIT Press, 2016.