COMPARISON OF EXTREME GRADIENT BOOSTING ALGORITHM AND ARTIFICIAL NEURAL NETWORK ON DIABETES PREDICTION

[1]Jevon Carla, [2]Y.b Dwi Setianto
[1,2]Program Studi Teknik Informatika Fakultas Ilmu Komputer,
Universitas Katolik Soegijapranata
[2]setianto@unika.ac.id

ABSTRACT

Diabetes is one of the serious diseases and it causes the sufferer to have high blood sugar due to the body unable to produce the required amount of insulin to regulate glucose. It may cause complications or may increase the risk of developing another disease like heart disease, kidney disease, blindness, etc. One of the best ways to fight this disease is by early diagnosis. If there are a lot of patient records, the machine learning classification algorithms play a great role in predicting whether a person has diabetes or not. The used dataset is Diabetes UCI Dataset from kaggle which has been collected using direct questionnaires from the patients of Sylhet Diabetes Hospital in Sylhet, Bangladesh, and approved by a doctor. The dataset has 520 data and 17 attributes. Several studies have been made in the last few decades and some of them show that Artificial Neural Networks (ANN) are one of the best algorithms for diabetes predictions, Extreme Gradient Boosting (XGBoost) is one of the popular machine learning algorithms used for classification, because of that reason the writer wants to find out whether XGBoost can be used on diabetes prediction and compare it with ANN. Both algorithms models were trained with the same ratio 80:20, 75:25, 70:30. 60:40, and 50:50. There are four models for the ANN with 3 hidden layers, 4 hidden layers, 5 hidden layers, and 6 hidden layers, as for the XGBoost models there are the first model with default parameters and the second one with the hyperparameters tuning. The accuracy, precision, recall, and f1 score of the models will be compared to find out which one has better performance. XGBoost performance able to achieve better performance but the third ANN models able to achieve highest score on 80:20, with 75:25 XGBoost with hyperparameters tuning able to achieve highest score, but XGBoost with default parameters have the same score as the the third ANN model, with 70:30 ratio, the third ANN model and both XGBoost models have the same score and have the highest score among all ratio. with 60:40 ratio, the first to third ANN models and XGBoost with default parameters have the same accuracy score but the third ANN models have the highest recall but lower precision than the XGBoost models. And with 50:50 XGBoost 2 has the best overall performances than the other models.

Keywords: diabetes, ann, xgboost, prediction, comparison

BACKGROUND

Diabetes is one of the chronic diseases in the world. It causes the sufferer to have high blood sugar or hyperglycemia due to the body unable to respond or produce the required amount of insulin to regulate the glucose in their body. It may cause complications, or it may increase the risk of developing into another disease such as heart disease, kidney failure, blindness, urinary organ diseases , etc. According to the World Health Organization (WHO) Report in 2016, everyone could be infected including children, women, men, young and old. Early fast diagnosis is one of the most important issues to fight this chronic disease . If there are a lot of patient records, the machine learning classification algorithms play a great role in predicting whether a person has diabetes or not [1].

The used dataset is Diabetes UCI Dataset from kaggle which has been collected using direct questionnaires from the patients of Sylhet Diabetes Hospital in Sylhet, Bangladesh, and approved by a doctor. The dataset has 520 data and 17 attributes. To make the performance efficient, the data was checked for missing values. After checked, the used dataset doesn’t have any missing values, label encoding was applied or transforming the instance of diabetes into numerical values e.g. 1 or 0 [2]. After that the dataset was divided into training data and testing data for both algorithms [3].

Several studies have been made in the last few decades and some of them show that Artificial Neural Networks (ANN) are one of the best algorithms for diabetes predictions and. XGBoost is one of the popular machine learning algorithms used for classification, because of that reason the writer wants to find out whether XGBoost can be used on diabetes prediction and comparing it with ANN [3], [7], [8].

LITERATURE STUDY

Sisodia et al. [10] presents classification algorithms such as Decision Tree, SVM and Naive Bayes to detect diabetes at an early stage. The authors use the Pima Indians Diabetes database on DT, SVM, NB algorithms for diabetes predictions. This research focuses on comparing the three algorithms and finding out which one has the best performance on diabetes prediction. This research will be useful for my research topic as the authors present three classification algorithms for diabetes algorithms. The limitations of this article are the research using a software tool called WEKA tool to perform the experiment which was designed by University of Waikato New Zealand. This article will not form the basis of my research; however it will be useful as supplementary information for my research on data pre-processing.

Sarwar et al. presents diabetes analytics prediction using KNN, Naive Bayes, SVM, Decision Tree, Logistic Regression and Random Forest [2]. The authors use data gained through National Institutes of Diabetes and DIgestive and Kidney diseases and dataset from Pima Indian from Kaggle. This article is useful for my research topic, as Sarwar et al. shows that feature selection plays an important role in prediction and is indicative of whether a patient will have diabetes or not. There are some limitations of the article such as the size of the dataset and missing attributes values, where to build a prediction model with 99,99% or the highest accuracy, we will need thousands of records without any missing values. Thus the author said we will need a lot of records without missing values to achieve the highest accuracy, in future work the author will focus on integration of other methods into the used model for tuning the parameters of models for better accuracy. Then testing those models with a large dataset with minimum or no missing values will reveal more insight and better accuracy. This article will be useful as supplementary information for my research for the mode and comparison.

Jasim et al. presents diabetes classification using K-Nearest Neighbor (K-NN) algorithm and compares it with artificial neural network (ANN) [1]. The authors use the Pima Indian dataset, in this study the authors perform the ANN and KNN by changing K value and number of hidden layers between 1 to 50. This research focuses on females of Pima Indians descent for diabetes predictions and comparing both classification methods in different ways. This research is useful for my research topic as Jasim et al. compare ANN and KNN which changing the number of hidden layers in ANN will increase the accuracy of ANN model. The main limitation of this study is it only focuses on females of Pima Indians descent who are at least 21 years old. The result of this study shows that ANN is better than KNN and by changing the number of hidden layers of ANN can increase its accuracy. This article will be useful as supplementary information for ANN models and comparison.

Sonar et al. presents diabetes prediction using different machine learning approaches namely Decision Tree, ANN, Naive Bayes, and SVM [6]. The authors use the Pima Indian Diabetes Dataset. Their research focuses on comparing those algorithms and analyzing which one has the best performance. The article is useful for my research topic as the research showed that SVM and ANN achieve the highest accuracy and suggest that ANN gives good predictions and it is easy to implement. The limitations are that ANN is difficult to handle big data with complex models and it also requires a huge amount of processing time. Thus the authors said so, SVM needs parameters to be set correctly, Decision Tree is relatively inaccurate and the NB is sensitive to how the inputs are prepared. This research will not form the basis of my research; however it will be useful supplementary information on ANN model.

Akter et al. present an early autism detection using improved machine learning techniques based on classification models [7]. The data used on this research were children , adolescent, and adults dataset from the University of California Irvine (UCI) and toddlers dataset from Kaggle and using different classifier algorithms. This research focuses on autism detection using different classifiers like ANN, RNN, Decision Tree, Gradient Boosting, KNN, Logistic Regression, MLP, SVM, Naive Bayes, Random Forest, and XGBoost to analyze and make comparisons of it. The article is useful to my research topic because it showed that ANN can/or has slightly better performance than XGBoost. The main limitation of this study is that the study focuses on autism detection but not diabetes. This article will not form the basis of my research; however it will be useful supplementary information for my research on comparison between XGBoost and ANN.

Hughest et al. presents a model of Parkinson’s disease patient gait using OLS, LASSO, XGBoost, ANN, and Symbolic Regression [3]. The authors used data obtained from the Gait in Parkinson’s Disease project on PhysioNet. This article focuses on the data from Frenkel-Toledo et al. , consisting of data from 64 subjects (35 PD patients and 29 controls). An Ultraflex Computer Dyno Graphy device with eight sensors was placed under each foot of the patient, and the patient walked unassisted on level ground in their usual manner, at a self-selected pace for two minutes and only the data from the 16 sensors was used in this study. The data was z-score normalized to enable an easier between subject comparison of the models being generated since the recorded data may not have been scaled in a meaningful way. This article is useful to my research as the authors compare XGBoost and ANN on this study. The main limitation of the article is that this study focuses on Parkinson's disease patient gait not diabetes, however the authors compare ANN and XGBoost in this study. The conclusion of this study shows that XGBoost fit the training data better and generalized to unseen data from the same subject better than ANNs, however the difference between model effectiveness at generalizing to unseen data from the same cohort was minimal. This article will not form the basis of my research, however it will be useful as a supporting article for comparison between XGBoost and ANN models on diabetes prediction.

Sun et al. [8] presents Alzheimer’s disease prediction using ANN and XGBoost and compares it. The dataset used in this article is from the Open Access Series of Imaging Studies (OASIS) , the longitudinal MRI data in non-demented and demented right-handed older adults aged 60-96. This article focuses on finding out whether XGBoost or ANN is better on Alzheimer’s disease prediction. This article is useful for my research as the authors compare ANN and XGBoost on disease prediction. There are few limitations on this study, firstly in the analysis of ANN, the missing value imputation uses the method of missForest, which is based on random forest. It has a probability of causing the overfitting of classification. Thirdly, the age group of the dataset focuses on 60-96, the analysis before 60 is not considered. The study shows that ANN method still has more error prediction than XGBoost, and in ANN, all missing values are firstly imputed by missForrest which based on Random Forest and to improve the accuracy rate the authors applied K-fold cross validation on both algorithms (K=10). This article will not form the basis of my research, however it will be useful as a supporting article for comparison between XGBoost and ANN models in diabetes prediction and supplementary article for improving the accuracy rate of the models using K-fold cross validation with K=10.

Srivastava et al. [9] presents a model to predict whether the patients suffer diabetes or not using ANN algorithms. The authors used the Pima Indians Diabetes Dataset from the Research Center of National Institute of Diabetes and Digestive and Kidney Diseases. It contains 8 attributes plus one class column. Each attribute is numeric-valued, and the data pre-processing is handling the missing value by replacing it with the mean of column method as the replace 0. This article focuses on diabetes prediction using the ANN model. This article is useful for my research as a supplementary article and reference of proposed ANN model and data pre-processing. The main limitation of this study is that the dataset has missing values and needs to be preprocessed before training it with the model. This article is useful for my research because the authors suggest that to fill the missing values , we can use the mean column method as the replacement. This article will not form the basis of my research, however it will be useful as a supplementary article and reference article for improving performance of ANN by data pre-processing.

Mubarok et al. presents hyperparameters tuning on xgboost model for heart failure prediction [12]. The authors used the dataset from Heart Failure Clinical Records, the authors using outlier removal to preprocess the dataset, then the dataset split into training and test set with ratio 80% and 20% respectively, then train the model without hyperparameters tuning, then applying random search, grid search, tree parzen estimator and compares the results, the hyperparameters will be validated using 5-Fold Cross Validation. The evaluation metric is AUC (Area Under the Curve), because the authors said that AUC is good for imbalanced data. The main limitation of this study is that this study focuses on heart failure. This article will not form the basis of my research, however it will be useful as a supplementary article and reference article for hyperparameters tuning on xgboost model.

Frimpong et al. presents diabetes type 2 prediction using feed forward neural network [15]. The authors use the Pima Indian Diabetes Dataset. The authors use accuracy, F1 Score, Recall, Precision as evaluation metrics, and the dataset was divided into 3 sets, namely training , testing, and validation with 70%, 20%, and 10% ratio respectively. The model consists of 3 hidden layers, the parameters are updated using Adam Optimizer with 0.001 of learning rate, with 500 batch size, and trained on 100 to 500 epochs. The limitations of this study are that this study didn't tell what data preprocessing had been done on the dataset before training it with the model. The conclusion of this study is that the proposed model effectively learns the attributes of the dataset , also avoiding underfit and overfitting issues , so there were less errors in the prediction of similar features or unknown test dataset. The authors also said the proposed technique makes the proposed model applicable to other medical datasets. This study will be useful as reference for my ANN model.

Ubaidillah et al. presents xgboost implementation with SMOTE and hyperparameters tuning with Bayesian Search on liver disease classification [16]. The used dataset is obtained from UCI Machine Learning Repository about Indian Liver Patient, the dataset has 583 instances and consists of 10 attributes. The authors apply the data preprocessing by removing the missing value, and label encoding to change string or non-numeric values into numeric values, for example in gender attribute there are two categories , male and female, then changing it into 0 and 1. Then the dataset is divided into training and testing with 80% and 20% ratio respectively. Then the dataset will be trained with 4 models, first one with default parameters or without hyperparameters tuning, second one with bayesian search, third one with smote and lastly the model were using SMOTE and bayesian search. The models were validated using 10-Folds validation to find the optimal hyperparameters. The main limitation of this study is that the study focuses on liver disease classification and the dataset needs to be cleaned first. The conclusion of this article is that missing values need to be cleaned to improve the accuracy of the models, and hyperparameters tuning on xgboost can increase its accuracy. This study will be useful as supplementary and references for my proposed xgboost model with hyperparameters tuning using bayesian search.

RESEARCH METHODOLOGY

Dataset

The used dataset of this study obtained from kaggle called Diabetes UCI Dataset which has been collected through direct questionnaires from the patients of Sylhet Diabetes Hospital in

Sylhet, Bangladesh, and approved by a doctor. It has total 520 data and 17 attributes including the 'class' attributes where Positive indicates the person had diabetes and Negative doesn't have diabetes. Table 1 shows the attributes information.

Table 1. Dataset UCI Diabetes Attributes Information

Attributes name	Details
Age	20-65
Sex	Male/Female
Polyuria	Yes/No
Polydipsia	Yes/No
sudden weight loss	Yes/No
weakness	Yes/No
Polyphagia	Yes/No
Genital thrush	Yes/No
visual blurring	Yes/No
Itching	Yes/No
Irritability	Yes/No
delayed healing	Yes/No
partial paresis	Yes/No
muscle stiffness	Yes/No
Alopecia	Yes/No
Obesity	Yes/No
Class	Positive/Negative

Data Preprocessing

Before training the dataset with the algorithms, data preprocessing was performed for the efficient performance of training [2], [4], [6]. It has a total of 520 and 17 attributes. During this preprocessing process, the dataset was checked if it has any missing values and the result is it has no missing values. And one hot encoding was performed where changing male into 1 and female 0, yes into 1 and no 2, and for the class were positive into 1 and negative into 0 [16]. Then splitting the dataset into X and Y where X is the attributes and Y is label or target, where Y is 'Class' that indicates either the person's positive or negative diabetes. And the dataset was divided into two different sets, namely for training and test purposes.

Artificial Neural Network (ANN)

Artificial Neural Network or ANN is a biological neural network model based machine learning algorithm. It is affected by the information that flows through the network since it learns through the input and output [4]. ANN general structure consists of input layer, hidden layer, and output layer. A process of adjusting the weights of all features by comparing the output value with the actual value in order to improve the precision of the model is the fundamental of ANN. The following is the basic learning process of ANN:

1. First, the perceptron receives the input values and applies the activation function to the output value while initializing the weights close to 0 [8].
2. Second, comparing the output value with the actual value and measuring the error in the prediction using the cost function [8].
3. Lastly, back propagation is applied to the neural network to adjust and update the information of weights to minimize the error in the second step, gradient descent is applied in this process to minimize the cost function. And then repeat from the first step until the const function is minimized [8].

Extreme Gradient Boosting (XG Boost)

Extreme gradient boosting or XGBoost is a tree boosting based scalable end-to-end algorithm which is used widely by data scientists to achieve state of the art results on many machine learning challenges. It was developed by Tianqi Chen and Carlos Guestrin[11]. It was introduced in 2014 and often used because of its speed, scalability, and effectiveness to solve classification or regression problems [12]. If the result shows high training accuracy, but low test accuracy, it is likely the model was overfitting, so to control the overfitting, a parameter tuning process is needed. Phan et al. study results showed that to demonstrate the superiority of the proposed XG Boost, parameter tuning is applied and it shows that the XGBoost model needs to be modified by the tuning parameter process [13].

In this study Bayesian Optimization is used, Hossain et al. shows that Bayesian Optimization is more efficient than Grid Search and Random Search because it can find the optimal combinations of hyperparameters by analyzing the previously tested values, and running the model based on previous tested values is much cheaper than running the whole objective function [14].

Evaluation Model

The evaluation will be seen by the accuracy rate, precision, recall, and F1 score whether XGBoost can be used or will have better results compared to ANN or not on diabetes prediction. In this study, the analysis is done by getting the accuracy, precision, recall and F1 score of both proposed algorithms and comparing them to find out which has the better result. Accuracy is the overall success rate of the model, the formula 1 below shows the formula of accuracy [2], [6], [15].

$$Accuracy = \frac{True\ Positive + True\ Negative}{Positive + Negative} \quad (1)$$

$$Precision = \frac{True\ Positive}{True\ Positive\ +\ False\ Positive} \quad (2)$$

$$Recall = \frac{True\ Positive}{True\ Positive\ + False\ Negative} \quad (3)$$

Precision is defined as the number of true positives compared to positive predictions, precision is shown as below [6], [15]. Recall is to measure the classifier completeness or the sensitivity, the formula 3 below shows the formula of recall [6], [15]. F1 score is the weighted average of the precision and recall, the formula 4 below shows the formula of F1 score [6], [15].

$$F1\ Score = 2 \times \frac{Precision \times\ Recall}{Precision\ +\ Recall} \quad (4)$$

After getting the accuracy, precision, recall, and f1 score of both algorithms, we compare it with each other to know if XGBoost can be used or has a better result than the ANN on diabetes prediction.

ANALYSIS

In this subchapter, the analysis of the study is explained. First, the algorithms will be trained and predict the test set. Second, calculate the accuracy, precision, recall, and F1 score from the predicted result, and compare the result of both algorithms.

DESIGN

In this subchapter, the design of the study is explained. First, load the dataset, the dataset was checked to see if there were any missing values, because it can lead to inaccurate results or reduce the model's accuracy [9]. The next step is data preprocessing, one hot encoding is applied where changing positive, male, and yes into 1 and negative, female, and no into 2. After data preprocessing is done, the dataset was split into X and Y, where Y is the label/target which contains the attribute "class" that indicates whether the person suffers diabetes or not. And next, split the dataset into 2 sets, namely training and test set, with ratio 80:20, 75:25, 70:30, 60:40, 50:50.

Second step, we train the dataset with both algorithms. First the Artificial Neural Network, the ANN will be built in three models. The first model consists of input layers, hidden layers and output layers. The input layer has 16 nodes indicating each feature or column of the dataset. Then three hidden layers are used, with 200 neurons for the first and second layer, and 150 neurons for the third layer. Rectified Linear Unit (ReLu) activation function is used for each layer. The model output layer has two nodes with Sigmoid activation function to determine either the person predicted with diabetes or not. binary cross entropy (BCE) loss is used to calculate the cost function. The neural network's parameters are updated using Adam optimizer with the learning rate of 0.001, and a batch size of 500. As for the used number of training or epochs is 500 [15]. After training the dataset with the ANN, then make the prediction with the test set and calculate for the analysis. The second model was built with 4 hidden layers and 200, 200, 200, 150 neurons respectively. The third model was built with 5 hidden layers and 200, 200, 150, 150, 150 neurons respectively. And the last one is 6 hidden layers with 200, 200, 200, 150, 150. 150 neurons respectively. All four models have the same Adam optimizer, learning rate, activation function, epochs, batch, input layer and output layers as the first ANN model.

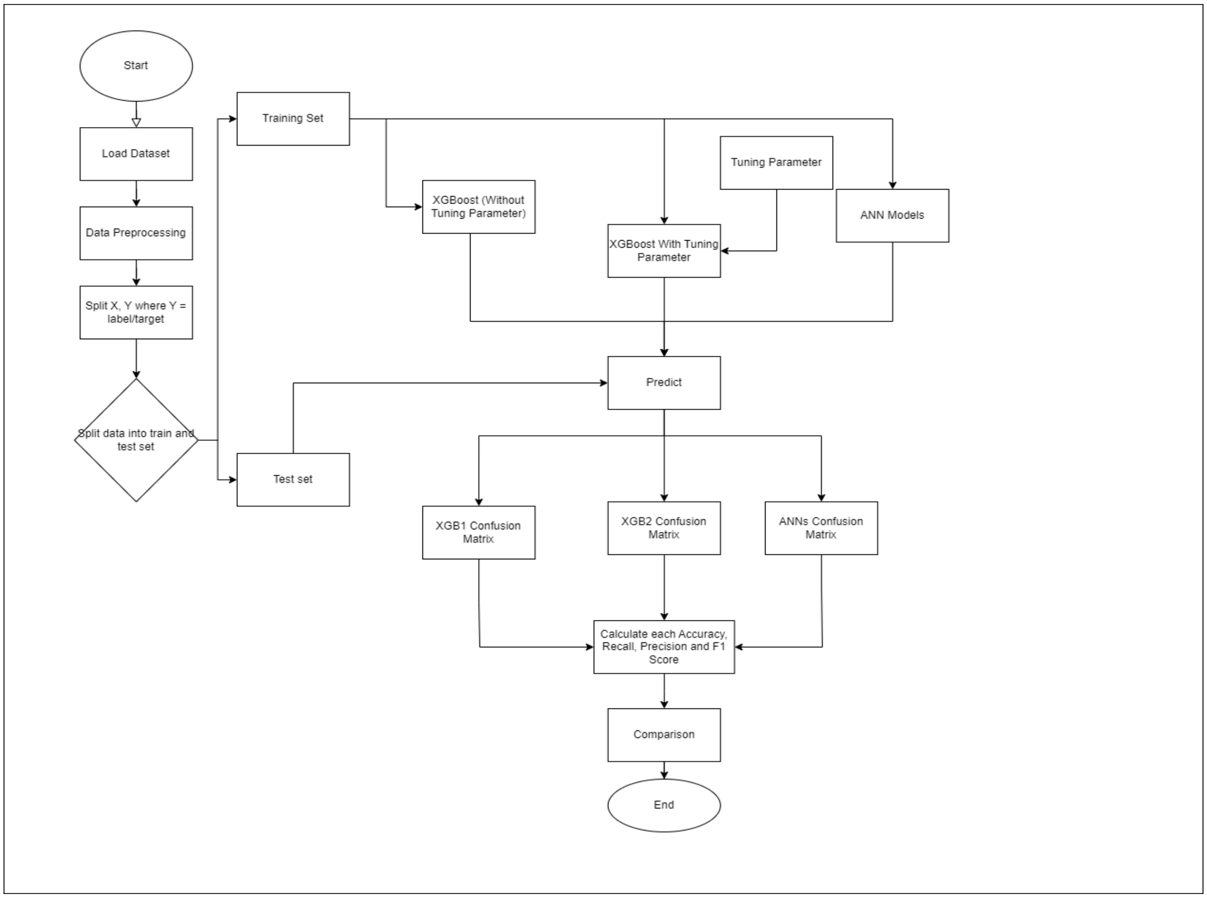

Figure 1. Flowchart of This Study

The second algorithm is the Extreme Gradient Boosting. First, the dataset was trained with the XGBoost without hyperparameters tuning, then tuning the hyperparameters using Bayesian Optimizer by applying 10-fold validation to the XGBoost and analyzing the result to find the best hyperparameters. The hyperparameters that tuned are learning_rate, min_child_weight, colsample_bytree, max_depth, gamma, n_estimators or the number of boosting rounds with ranged as following Table 2.

The used evaluation metric is Area Under Curve (AUC), because it was one of the best ways to get the classification model performance result generally, the unbalanced class, where the higher AUC score shows the best performance. [16]. After getting the best parameters, then the dataset is trained with the tuned parameters XGBoost , and makes a prediction with the test set. Lastly calculate the prediction result for the analysis.

After training, makes predictions and calculates for the analysis with both algorithms. We compare both algorithms accuracy, precision, recall, and f1 scores. From the comparison then we can find out which algorithms have the better performance or results on diabetes prediction.

Table 2. Hyperparameters Search Spaces

Hyperparameters	**Range**
max_depth	$3 - 11$
gamma	$1 \times 10^{-9} - 0.5$
n_estimators	$100 - 250$
learning_rate	$0.01 - 0.5$
min_child_weight	$1 - 10$
colsample_bytree	$0.1 - 0.8$

RESULTS AND DISCUSSION

In this subchapter, the results of the models will be compared. ANN 1 indicates the first ann model that has 3 hidden layers with 200, 200, and 150 neurons respectively, ANN 2 indicates the second ann model which has 4 hidden layers with 200, 200, 200, and 150 neurons respectively. ANN 3 indicates the third ann model which has 5 hidden layers with 200, 200, 150, 150, and 150 neurons respectively. ANN 4 indicates the fourth ann model which has 6 hidden layers with 200, 200, 200, 150, 150, and 150 neurons respectively. XGBoost 1 is the first xgboost model without hyperparameters tuning or with the default parameters [17], and XGboost 2 is the second model with hyperparameters tuning.

Table 3. Results Comparison with Train set 80% and Test set 20%

Model	**Recall**	**Precision**	**F1 Score**	**Accuracy**
ANN 1	0.958	0.986	0.971	0.962
ANN 2	0.958	0.986	0.971	0.962
ANN 3	0.986	0.986	0.986	0.981
ANN 4	0.972	0.986	0.979	0.971
XGBoost 1	0.958	1.0	0.978	0.971
XGBoost 2	0.958	1.0	0.978	0.971

Table 3 shows that with ratio 80:20, ANN 3 has the best overall performance of all models, followed by ANN 4, XGBoost 1 and XGBoost 2. ANN 4 and the XGBoost models have the same accuracy score but ANN 4 have better recall and F1 score, even though the precision is better on the XGBoost models.

Table 4. Results Comparison with Train set 75% and Test set 25%

Model	Recall	Precision	F1 Score	Accuracy
ANN 1	0.952	0.988	0.97	0.962
ANN 2	0.952	1.0	0.976	0.969
ANN 3	0.964	0.988	0.976	0.969
ANN 4	0.976	0.863	0.916	0.885
XGBoost 1	0.952	1.0	0.976	0.969
XGBoost 2	0.964	1.0	0.982	0.977

Table 4 shows that XGBoost 2 has the best performances over all models, followed by ANN 2 , ANN 3 and XGBoost 1. ANN 2 and XGBoost 1 have lower Recall than the ANN 3 but they have the same F1 score and have better precision.

Table 5. Results Comparison with Train set 70% and Test set 30%

Model	Recall	Precision	F1 Score	Accuracy
ANN 1	0.961	0.99	0.975	0.968
ANN 2	0.912	0.979	0.944	0.929
ANN 3	0.98	1.0	0.99	0.987
ANN 4	0.98	0.99	0.985	0.981
XGBoost 1	0.98	1.0	0.99	0.987
XGBoost 2	0.98	1.0	0.99	0.987

Table 5 shows that with 70% of the training set and 30% of the test set, both XGBoost models and ANN 3 achieve the same result and have the best performances, followed by ANN 4, ANN 1 and the last is ANN 2.

Table 6. Results Comparison with Train set 60% and Test set 40%

Model	Recall	Precision	F1 Score	Accuracy
ANN 1	0.939	0.992	0.965	0.957
ANN 2	0.939	0.992	0.965	0.957
ANN 3	0.947	0.984	0.965	0.957
ANN 4	0.939	0.939	0.939	0.923
XGBoost 1	0.932	1.0	0.965	0.957
XGBoost 2	0.924	1.0	0.961	0.952

Table 6 shows that with 60% of the training set and 40% of the test set, ANN 1, ANN 2, ANN 3, and XGBoost 1 able to achieve the same accuracy score. ANN 1 has the same results as the ANN 2, but has lower recall , precision than the ANN 3. XGBoost 1 has a lower recall than the three ann models mentioned.

Table 7. Results Comparison with Train set 50% and Test set 50%

Model	Recall	Precision	F1 Score	Accuracy
ANN 1	0.91	1.0	0.953	0.942
ANN 2	0.904	1.0	0.949	0.938
ANN 3	0.934	0.994	0.963	0.954
ANN 4	0.795	0.971	0.874	0.854
XGBoost 1	0.94	1.0	0.969	0.962
XGBoost 2	0.946	1.0	0.972	0.965

Table 7 shows that with 50% of the training set and with 50% of the test set, XGBoost 2 has the best overall performance, followed by XGBoost 1, ANN 3, ANN1, and ANN 4.

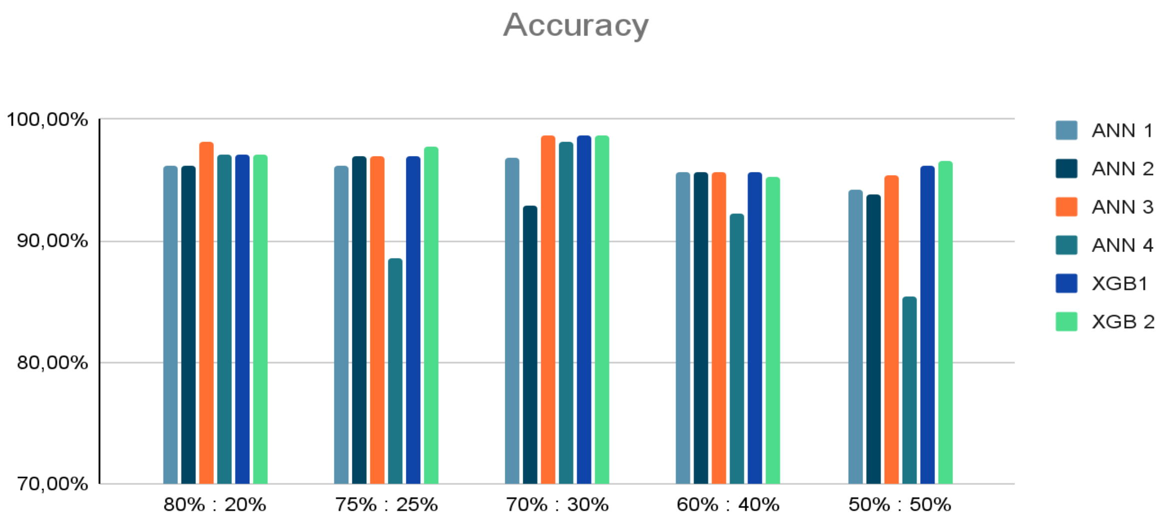

Figure 2. Accuracy Comparison

CONCLUSION

Based on the result, XGBoost can be used on diabetes prediction, and it is able to achieve accuracy above 90% which is good. So, based on the result, both models are usable and good for diabetes prediction. From Table 3 to 7 and Figure 2 shows that with 70% ratio of training set is the best ratio for this dataset, because ANN 1, ANN 3, ANN 4, XGBoost 1 and XGBoost 2 achieve the highest score compared with different ratio. ANN 3 is able to achieve better accuracy scores at 80% of training data and achieve the same accuracy score at 70% of training data as the XGBoost models. Some of the ANN models are able to achieve same result as the XGBoost models, as we can see with 80%:20% ratio, ANN 4 have the same accuracy score as the XGBoost models but it has better recall and f1 score. With 60%:40%, ANN 1, ANN 2, ANN 3, and XGBoost 1 able to achieve the same accuracy scores, but ANN 3 have the highest recall but lower precision than ANN 1, ANN 2, and XGBoost 1, as for XGBoost 2 is second lowest accuracy score. And the last with 50% of training data XGBoost 2 have the best overall performance. XGBoost with hyperparameters tuning (XGBoost 2) able to achieve the same result or better performance than XGBoost with default parameters (XGBoost 1), the results show that with 75% or 50% of training data XGBoost 2 outperforms XGBoost 1 but with 60% of training data XGBoost 1 have better performance than XGBoost 2. And with 80% ANN 3 able to outperform the XGBoost models.

So, with lower ratio of training set XGBoost are better but with higher ratio of training set ANN 3 able to achieve better result than the XGBoost models. The ANN models were built with 3, 4, 5, 6 hidden layers respectively, but the model with 5 hidden layers give the best performances over all ANN models, the last model's performances began decreasing or have big gap than the previous with 5 hidden layers. by adding more hidden layers we also can improve the ANNs performances but it's only to certain extend, after that the performance began to drop. We can improve the XGBoost models by adding more data to the training set and hyperparameters tuning to get the optimal parameters.

REFERENCES

[1] Jasim, I. S., Deniz Duru, A., Shaker, K., Abed, B. M., & Saleh, H. M. (2017). Evaluation and measuring classifiers of diabetes diseases. 2017 International Conference on Engineering and Technology (ICET), 1–4. https://doi.org/10.1109/ICEngTechnol.2017.8308165

[2] Sarwar, M. A., Kamal, N., Hamid, W., & Shah, M. A. (2018). Prediction of Diabetes Using Machine Learning Algorithms in Healthcare. 2018 24th International Conference on Automation and Computing (ICAC), 1–6. https://doi.org/10.23919/IConAC.2018.8748992

[3] Hughes, J. A., Houghten, S., & Brown, J. A. (2020). Models of Parkinson's Disease Patient Gait. IEEE Journal of Biomedical and Health Informatics, 24(11), 3103–3110. https://doi.org/10.1109/JBHI.2019.2961808

[4] Zhang, Y., Lin, Z., Kang, Y., Ning, R., & Meng, Y. (2018). A Feed-Forward Neural Network Model For The Accurate Prediction Of Diabetes Mellitus. 7(8), 5. https://www.ijstr.org/final-print/aug2018/A-Feed-forward-Neural-Network-Model-For-The-Accurate-Prediction-Of-Diabetes-Mellitus-.pdf

[5] Lakhwani, K., Bhargava, S., Hiran, K. K., Bundele, M. M., & Somwanshi, D. (2020). Prediction of the Onset of Diabetes Using Artificial Neural Network and Pima Indians Diabetes Dataset. 2020 5th IEEE International Conference on Recent Advances and Innovations in Engineering (ICRAIE), 1–6. https://doi.org/10.1109/ICRAIE51050.2020.9358308

[6] Sonar, P., & JayaMalini, K. (2019). Diabetes Prediction Using Different Machine Learning Approaches. 2019 3rd International Conference on Computing Methodologies and Communication (ICCMC), 367–371. https://doi.org/10.1109/ICCMC.2019.8819841

[7] Akter, T., Khan, Md. I., Ali, M. H., Satu, Md. S., Uddin, Md. J., & Moni, M. A. (2021). Improved Machine Learning based Classification Model for Early Autism Detection. 2021 2nd International Conference on Robotics, Electrical and Signal Processing Techniques (ICREST), 742–747. https://doi.org/10.1109/ICREST51555.2021.9331013

[8] Sun, X. (2021). Application and Comparison of Artificial Neural Networks and XGBoost on Alzheimer's Disease. Proceedings of the 2021 International Conference on Bioinformatics and Intelligent Computing, 101–105. https://doi.org/10.1145/3448748.3448765

[9] Srivastava, S., Sharma, L., Sharma, V., Kumar, A., & Darbari, H. (2019). Prediction of Diabetes Using Artificial Neural Network Approach. In K. Ray, S. N. Sharan, S. Rawat, S. K. Jain, S. Srivastava, & A. Bandyopadhyay (Eds.), Engineering Vibration, Communication and Information Processing (Vol. 478, pp. 679–687). Springer Singapore. https://doi.org/10.1007/978-981-13-1642-5_59

[10] Sisodia, D., & Sisodia, D. S. (2018). Prediction of Diabetes using Classification Algorithms. Procedia Computer Science, 132, 1578–1585. https://doi.org/10.1016/j.procs.2018.05.122

[11] Chen, T., & Guestrin, C. (2016). XGBoost: A Scalable Tree Boosting System. Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 785–794. https://doi.org/10.1145/2939672.2939785

[12] Mubarok, M. R., & Herteno, R. (2022). HYPER-PARAMETER TUNING PADA XGBOOST UNTUK PREDIKSI KEBERLANGSUNGAN HIDUP PASIEN GAGAL JANTUNG. 09, 11.

[13] Phan, Q.-T., Wu, Y.-K., & Phan, Q.-D. (2020). A Comparative Analysis of XGBoost and Temporal Convolutional Network Models for Wind Power Forecasting. 2020 International Symposium on Computer, Consumer and Control (IS3C), 416–419. https://doi.org/10.1109/IS3C50286.2020.00113

[14] Hossain, R., & Timmer, D. D. (2021). Machine Learning Model Optimization with Hyper Parameter Tuning Approach. 8.

[15] Frimpong, E. A., Oluwasanmi, A., Baagyere, E. Y., & Zhiguang, Q. (2021). A feedforward artificial neural network model for classification and detection of type 2 diabetes. Journal of Physics: Conference Series, 1734(1), 012026. https://doi.org/10.1088/1742-6596/1734/1/012026

[16] Ubaidillah, R., Muliadi, M., Nugrahadi, D. T., Faisal, M. R., & Herteno, R. (2022). Implementasi XGBoost Pada Keseimbangan Liver Patient Dataset dengan SMOTE dan Hyperparameter Tuning Bayesian Search. JURNAL MEDIA INFORMATIKA BUDIDARMA, 6(3), 1723. https://doi.org/10.30865/mib.v6i3.4146

[17] Budholiya, K., Shrivastava, S. K., & Sharma, V. (2022). An optimized XGBoost based diagnostic system for effective prediction of heart disease. Journal of King Saud University - Computer and Information Sciences, 34(7), 4514–4523. https://doi.org/10.1016/j.jksuci.2020.10.013