Diabetes Mellitus Prediction using Different Ensemble Machine Learning Approaches

Md. Tanvir Islam[1], M. Raihan[2], Nasrin Aktar[3], Md. Shahabub Alam[4], Romana Rahman Ema[5] and Tajul Islam[6]
Department of Computer Science and Engineering, North Western University, Khulna, Bangladesh[1-3,5,6]
Khulna University of Engineering & Technology, Khulna, Bangladesh[1,5,6]
Ahsanullah University of Science and Technology, Dhaka, Bangladesh[4]
Emails: tanvirislamnwu@gmail.com[1], mraihan@nwu.edu.bd[2], raihanbme@gmail.com[2], nasrinlipinwu@gmail.com[3], nabid.aust37@gmail.com[4], romanacsejstu@gmail.com[5] and tajulkuet09@gmail.com[6]

***Abstract*—Nowadays Diabetes Mellitus is one of the most rapidly growing diseases which makes the biggest contribution to morbidity and mortality worldwide. Diabetes Mellitus is a group of metabolic disorders defined by high blood glucose level over a prolonged period. Although this disease is familiar as hereditary disease, many people are suffering from this disease without having family background. If diabetes is not in control, the level of glucose goes up and it may cause damage to small vessels in human body which appears most often in the nerves, feet, eyes even in heart and kidneys. To get rid of these issues, it is very crucial to predict diabetes on the early stage. Hence, we have decided to do research on diabetes prediction using Machine Learning algorithms. In this study, we have used three popular Machine Learning algorithms called AdaBoost, Bagging and Random Forest. To train and test the algorithms we have collected real time information of both diabetic and non-diabetic people. The dataset contains 464 instances with 22 unique risk factors. In between the three algorithms, AdaBoost gave 97.84% accuracy, Bagging gave 98.28% accuracy and Random Forest gave 99.35% accuracy with respect to predict diabetes disease precisely.**

***Keywords*—Diabetes Mellitus, Machine Learning, Classification, Prediction, AdaBoost, Bagging, Random Forest.**

I. Introduction

Diabetes Mellitus (DM) is generally known as Diabetes which blocks human body from getting the energy properly from the food we eat. It is a chronic stage associated with unusually high level of glucose in our blood. The pancreas produces insulin which lowers the glucose level. The insufficiency of production of insulin or any inability in using insulin properly in our body causes diabetes [1]. DM has been one of the fastest spreading diseases at present world. According to statistics, by the end of 2017, approximately 425 million people aged between 20 to 79 years were having diabetes and it is estimated that this number will rise to 629 million before 2045 [2]. By 2015, 30.1 million or 9.4% of Americans were affected by diabetes, among them 1.25 million were children [3]. Every year 1.5 million new affected Americans are joining in this list. Among the mature people in the top five South-East Asian countries, Bangladesh was the second in the list with 5.2 million DM patients in 2013. It is estimated that this number will rise to 8.20 million in 2035 [4]. So, it is clear that, DM has been a universal problem and it is high time to find out the best practical solution. Machine Learning (ML) is the field of Data Mining and study of algorithms where these types of problems can be solved using algorithms and sample datasets [5].

The motive of our work is to analysis on diabetes patients' datasets to recognize diabetes accurately using three ML algorithms, AdaBoost, Bagging and Random Forest (RF).

II. Related Works

Ayman Mir et al. [6] have performed an analysis to predict diabetes disease using ML techniques on big data of healthcare. They used several ML algorithms such as Naïve Bayes, Support Vector Machine (SVM), Random Forest and Simple CART. The dataset contains 9 attributes with having both numerical and nominal values. The obtained accuracy for Naïve Bayes is 77%, SVM is 79.13%, RF is 76.5%, and Simple CART is 76.5%. Another research performed for the purpose of indicating the critical features for predicting diabetes. The algorithms have been used in this research are Logistic Regression (LR), SVM and RF. In the analysis, researchers found RF as the best algorithm to predict diabetes which gave 84% accuracy [7]. Similarly, an analysis has been conducted based on ML algorithms where analysists used SVM, AdaBoost, Bagging, K-NN, RF algorithms with a dataset of 506 instances and 30 features. They got 75.49% accuracy for AdaBoost, 76.28% for Bagging, 72.33% for K-NN, 75.30% for RF, 72.72% for SVM [8]. Durga Kinge et. al. conducted an analysis to determine the performances of several algorithms named Decision Tree (J48), Naïve Bayes, RF, AdaBoost, Bagging, Multilayer Perceptron (MLP), Simple Logistic to predict diseases using data mining and ML techniques. A dataset of heart disease having total 303 instances with 74 raw attributes was taken and only one 14 significant features were used among them. Accuracy for J48, Naïve Bayes, RF, AdaBoost, Bagging, Multilayer Perceptron (MLP), Simple Logistic algorithms are 78.15%, 82.59%, 83.15%, 81.59%, 81.59%, 79.41% and 83.1% respectively [9]. Soumayadeep Manna et al. have conducted a research to predict the important factors that cause diabetes. They have used a dataset which contains 3075 instances and each instance has 8 factors. They have used LR and RF whereas RF gave 86.70% accuracy and LR gave 89.17% accuracy [10]. Deepika Verma and Nidhi Mishra conducted a study to identify DM by using a dataset on Naïve Bayes, J48, Sequential Minimal Optimization (SMO), MLP, and Reduces Error Pruning Tree (REP-tree) algorithms and they found SMO to give 76.80% accuracy on diabetes dataset [11]. Another research team developed a system using

RF algorithm based on some variables like age, weight, hip, waist, height etc. They performed the analysis based on 4 groups of datasets. The proposed system gave 84.19% accurate result in terms of prediction [12].

The objective of our research is to find out new risk factors of diabetes patients' and predict diabetes mellitus accurately using three ML algorithms, AdaBoost, Bagging and Random Forest (RF).

III. Methodology

We have chosen three popular Ensemble Machine Learning algorithms. Ensemble Learning is a strong way to increase the performance of a model. It works based on multiple learning algorithms. It combines the outcomes of the learning algorithms to generate an optimal result. Thus, Ensemble ML algorithm is efficient to produce an accurate result in terms of prediction.

This study can be divided in the following sections:

- Data Collection
- Data Preprocessing
- Data Training
- Application of algorithms
- Tools

The flowchart in Fig. 1 shows the whole working procedure of the study.

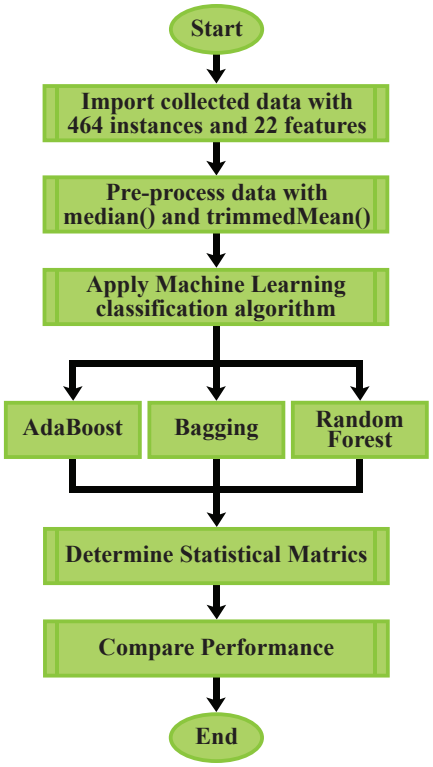

Fig. 1: Flowchart of the whole work procedure

TABLE I: Features list

Feature Names	Subcategory	Data Distribution Mean, Median
Age	L.V. = 20 yrs	41.65, 40.40
	H.V. = 83 yrs	
Gender	Male	48.92%
	Female	51.08%
Drug history	Yes	57.11%
	No	42.89%
Weight Loss	Yes	46.34%
	No	53.66%
Diastolic blood pressure (BP)	L.V. = 80 mmHg	117.8, 120
	H.V. = 170 mmHg	
Systolic blood pressure(BP)	L.V. = 50 mmHg	77.67, 80
	H.V. = 110 mmHg	
Duration of diabetes	L.V. = 0 year	713.8, 90
	H.V. = 20 years	
Height	L.V. = 138 cm	155.9, 156
	H.V. = 174 cm	
Weight	L.V. = 37 kg	60.79, 60
	H.V. = 85 kg	
Blood sugar before eating	L.V. = 3.07 mmol/L	7.792, 7.205
	H.V. = 20.6 mmol/L	
Blood sugar after eating	L.V. = 5.8 mmol/L	13.003, 12.760
	H.V. = 28.09 mmol/L	
Urine color before eating	Nil	46.55%
	Blue	10.13%
	Yellow	9.70%
	Orange	15.73%
	Green	16.81%
	Brick Red	0.43%
	Green Yellow	0.65%
Urine color after eating	Nil	46.98%
	Blue	6.90%
	Red	2.80%
	Yellow	9.27%
	Orange	22.20%
	Green	7.11%
	Brick Red	3.23%
	Green Yellow	1.51%
Waist	H.V. = 28 cm	35.22, 35.00
	L.V. = 44 cm	
Thirst	Yes	48.06%
	No	51.94%
Hunger	Yes	44.61%
	No	55.39%
Relatives	Yes	58.41%
	No	41.59%
Pain or numbness	Yes	46.55%
	No	53.45%
Blurred vision	Yes	53.45%
	No	46.55%
Type of medicine	No	35.99%
	Tablet	37.07%
	Insulin	26.94%
Family stroke	Yes	42.89%
	No	57.11%
Physical activity	Yes	90.30%
	No	9.70%
Diabetes Mellitus (Class)	Yes	65.09%
	No	34.91%
L.V. = Lowest Value		
H.V. = Highest Value		

A. Data Collection

For collecting data, we went to several diagnostic centers in Khulna, Bangladesh. In our dataset, there are total 464 instances, where 237 are female and 227 male patients. Each instance has 23 attributes including the outcome class and 22 attributes excluding the outcome class which make our dataset more specific. Name of those attributes are written in Table I.

B. Data Preprocessing

The collected dataset had a number of missing information. We used two functions from R-3.5.3 to manage those missing

information. We have used trimmedMean() to drop a certain percentage of the minimum and maximum observations and take the mean of the rest scores the dataset [13] and its outcome vary from the outcome of the standard R function mean() [14]. The other one is median() to find out the exact middle value of our dataset [15].

C. Data Training

We have applied percent split in order to splitting the dataset into two sub classes categorized by our training data and the test data or validation data. The ratio of training and test data in our dataset is 7:3.

D. Application of Algorithms

We have implemented three Ensemble Machine Learning algorithms. They are:

1) Adaboost: AdaBoost or *Adaptive Boosting* was the first practical successful boosting algorithm. It was developed for binary classification. It spotlights on classification related problems and focus on conversion of a set of weak classifiers into stronger one. AdaBoost works based on decision trees. Since, the trees are very short and only includes one decision for classification and they are sometimes called decision stumps. In the training dataset, every single instance is weighted which is initially set to [17],

$$weight(x_i) = \frac{1}{n}$$

Where, x_i = i^{th} training instance n = number of training instances The final classification equation can be represented as,

$$F(x) = sign(\sum_{m=1}^{M} \Theta_m f_m(x)),$$

Where,

$$f_m = m^{th}\, weak\, classifier$$

$$\theta_m = corresponding\, weight$$

As a matter of fact, it is the weighted amalgamation of M frail classifier.

2) Bagging: Bagging is one of the main two subdivisions of Ensemble Machine Learning algorithms. The main purposes of using Bagging are classification and regression. It can also be used with decision trees, where it upgrades the stability of a model by upgrading the accuracy and decreasing variance, thus decreasing the problem of overfitting [16]. Suppose, we have L almost independent dataset of size B denoted.

$$\left\{z_1^1, z_2^1, ..., z_B^1\right\}, \left\{z_1^2, z_2^2, ..., z_B^2\right\}, ..., \left\{z_1^L, z_2^L, ..., z_B^L\right\}$$

$$z_b^1 \equiv b^{th}\, observation\, of\, the\, l^{th}\, bootstrap\, sample$$

We can fit only L almost free weak learners,

$$w_1(.), w_2(.), ..., w_L(.)$$

Then, aggregate these into two averaging process to get an ensemble model,

$$s_L(.) = \frac{1}{L}\sum_{l=1}^{L} w_l(.)$$

(simple average, for regression problem)

$$s_L(.) = arg_k max\, [card(l \mid w_l(.) = k)]$$

(simple majority vote, for classification problem)

There are many ways to aggregate the dynamic models fitted in parallel. Bagging creates a number of sub-samples randomly of a dataset with replacement. Next, it trains a CART model on every sample. Next, after given a new dataset, it calculates the average prediction from each model.

3) Random Forest: Today, Random Forest is one of the versatile supervised classification algorithm of Machine Learning. We can observe it from its name to create a forest by several ways and then make it random. There is a direct connection between the tree number of the forest and the result we get through it. The more trees in the forest, the more accurate result we get [18]. To use Random Forest in regression problems, we use Mean Squared Error (MSE) to see how our data branches from nodes.

$$MSE = \frac{1}{N}\sum_{i=1}^{N}(f_i - y_i)^2$$

Where, N is the number of data points, f_i is the value returned by the model and y_i is the actual value of data point i.

When, we use Random Forest algorithm in classification problems, we should know that we often apply Gini index or the method to decide how the nodes on a decision tree branch.

$$Gini = 1 - \sum_{i=1}^{C}(p_i)^2$$

This method uses class and probability to find out the Gini of every branch on node and determine the best possible branch likely to occur [18].

We can also use entropy for determining the way node branches in a decision tree.

$$Entropy = \sum_{i=1}^{C} -p_i * log_2(p_i)$$

Thus, RF algorithm works.

E. Simulation Software and Environment

- R-3.5.3
- RStudio 1.1.463

IV. Outcomes

The results of the analysis have been analyzed based on a number of performance parameters given below:

A. Accuracy:

It is also known by Correctly Classified Instances [19].

$$Accuracy = \frac{T_p + T_n}{T_p + T_n + F_p + F_n}$$

B. Sensitivity/True Positive (TP) Rate:

It estimates the proportion of genuine positives that are accurately distinguished [20].

$$TPR = \frac{T_p}{P}$$

C. Precision (PRE):

It can be defined as [20],

$$PRE = \frac{T_p}{T_p + F_p}$$

D. F1 Score:

It is also known by F-Measure which can be denoted as,

$$FM = 2 \times \frac{PRE \times REC}{PRE + REC} = \frac{2 \times T_p}{2 \times T_p + F_p + F_n}$$

Where, FM = F1 Score / F-measure

E. False Positive Rate (FPR):

If a system or model predict a positive class incorrectly then the result is called False Positive which is also known as Fall-Out [19].

$$FPR = \frac{F_p}{F_p + T_n}$$

F. False Negative Rate:

If a model predicts negative class incorrectly then it is known as False Negative Rate. It is also known by Miss Rate [19].

G. MCC:

The full meaning of MCC is Matthews Correlation Coefficient which is the cohesion between PRE and REC [20].

TABLE II: Outcomes of AdaBoost algorithm

AdaBoost	
Precision	0.9750
Sensitivity	96.30%
Specificity	98.68%
F1 Score	0.9689
False Positive Rate	0.0132
False Negative Rate	0.0250
False Discovery Rate	0.0247
Negative Predictive Value	0.9803
Matthews Correlation Coefficient	0.9525
Accuracy	97.84%

TABLE III: Outcomes of Bagging algorithm

Bagging	
Precision	0.9753
Sensitivity	97.53%
Specificity	98.68%
F1 Score	0.9753
False Positive Rate	0.0132
False Negative Rate	0.0247
False Discovery Rate	0.0247
Negative Predictive Value	0.9868
Matthews Correlation Coefficient	0.9621
Accuracy	98.28%

H. Explanation of the Analysis:

The outcomes of this study have been described below:

Table II illustrates the outcomes of performance variables of AdaBoost algorithm. We have got Precision 0.9750, Sensitivity and Specificity are 96.30% and 98.68% respectively. Again, F1 score has come 0.9689, while False Positive Rate is 0.0132, False Negative Rate is 0.0250 and False Discovery Rate is 0.0247. Negative Predictive Value is 0.9803 as well as Mathews Correlation Coefficient is 0.9525. Overall accuracy is 97.84% when we use AdaBoost algorithm.

Table III displays the results of Bagging algorithm where we have got Precision 0.9753 while Sensitivity and Specificity are 97.53% and 98.68% respectively. FI score or F-measure is 0.9753. In addition, False Positive Rate is 0.0132, False Negative Rate is 0.0247 and False Discovery Rate is 0.0247. Again, the Negative Predictive Value is 0.9868 and Matthews Correlation coefficient is 0.9621. We have received 98.28% accuracy for Bagging algorithm which is more than the accuracy we have got from AdaBoost algorithm.

Table IV shows the outcomes of performance parameters of Random Forest algorithm. We have got 99.35% accuracy RF algorithm which is really impressive while the value of Precision is 0.9877. Furthermore, Sensitivity and Specificity

TABLE IV: Outcomes of Random Forest algorithm

Random Forest	
Precision	0.9877
Sensitivity	99.38%
Specificity	99.34%
F1 Score	0.9907
False Positive Rate	0.0066
False Negative Rate	0.0062
False Discovery Rate	0.0123
Negative Predictive Value	0.9967
Matthews Correlation Coefficient	0.9858
Accuracy	99.35%

TABLE V: Outcomes comparison of three algorithms

Evaluation Metrics	Machine Learning Algorithms		
	AdaBoost	Bagging	Random Forest
Precision	0.9750	0.9753	0.9877
Sensitivity	96.30%	97.53%	99.38%
Specificity	98.68%	98.68%	99.34%
F1 Score	0.9689	0.9753	0.9907
False Positive Rate	0.0132	0.0132	0.0066
False Negative Rate	0.0250	0.0247	0.0062
False Discovery Rate	0.0247	0.0247	0.0123
Negative Predictive Value	0.9803	0.9868	0.9967
Matthews Correlation Coefficient	0.9525	0.9621	0.9858
Accuracy	97.84%	98.28%	99.35%

are 99.38% and 99.34% respectively for RF. Using Random Forest, the F1 Score has come 0.9907 while False Positive Rate is 0.0066, False Negative Rate is 0.0062 and False Discovery Rate is 0.00123. Besides, the Negative Predictive Value is 0.9967 and the value of Mathews Correlation Coefficient is 0.9858.

Table V displays the comparison among the three ML algorithms, AdaBoost, Bagging and RF based on several evaluation metrics. We have got the highest accuracy 99.35% using Random Forest algorithm while AdaBoost and Bagging gave 97.84% and 98.28% accuracy respectively. Besides, using Random Forest, we have got 99.38% Sensitivity and 99.34% Specificity. In contrast, the Sensitivity and Specificity for AdaBoost are 96.30% and 98.68% while for Bagging it's 97.53% and 98.68% respectively. So, overall among the three algorithms, Random Forest performs better than AdaBoost and Bagging.

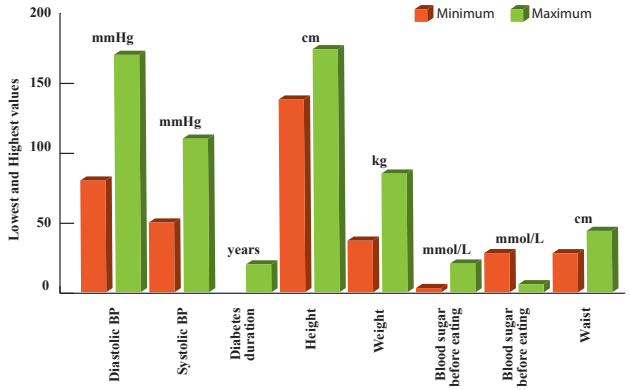

Fig. 2: Minimum and Maximum values for several numerical attributes

Fig. 2 represents the lowest and the highest values for some numerical attributes such as, the lowest Diastolic BP among the patients we have found 80 mmHg and the highest is 170 mmHg. The maximum height is 180 cm and the minimum height is 140 cm.

Fig. 3 displays mean and median for some numerical attributes for example, mean of duration of diabetes is 713.8 and median is 90. Again, both mean and median of height are same, 156 cm.

Fig. 4 exposes the percentages of urine color of diabetic

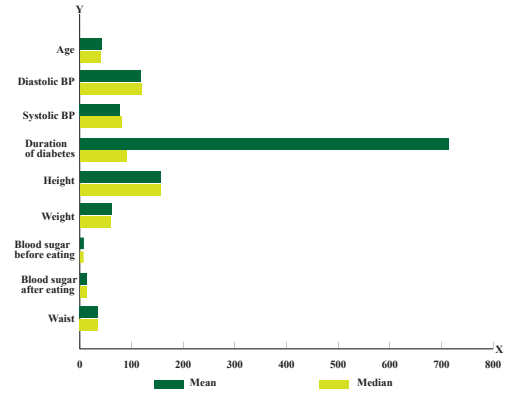

Fig. 3: Mean and Median values for several numerical attributes

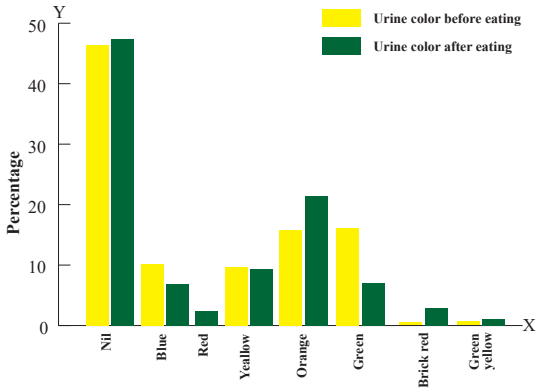

Fig. 4: Percentages of urine color of before and after eating

patients of before and after eating. For instance, before eating 15.73% patients urine color is orange while after eating it increases to 22.20%.

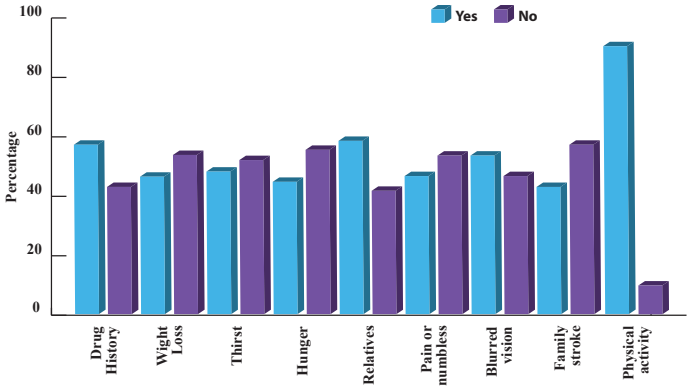

Fig. 5: Percentages between positive and negative aspects according to some nominal attributes

Fig. 5 reveals the percentages of the nominal variables of the dataset. For instance, according to the figure 57.11% patients have drug history, rest haven't taken any drugs. Again, 90.30% do perform between physical activity regularly while 9.70% don't.

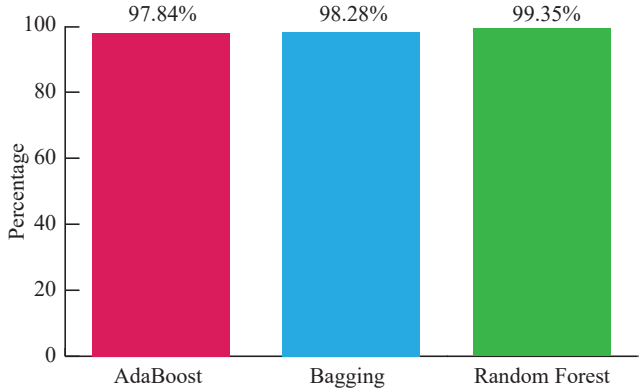

Fig. 6: Comparison between algorithms based on their accuracy

TABLE VI: Comparison between our proposed system and several existing systems

Reference No.	No. of Features	Sample Size	Algorithms	Accuracy
[6]	9	768	Random Forest	76.5%
[7]	9	768	Random Forest	84%
[8]	30	506	AdaBoost	75.49%
			Bagging	76.28%
			Random Forest	75.30%
[9]	14	303	AdaBoost	81.59%
			Bagging	81.59%
			Random Forest	83.15%
[10]	8	3075	Random Forest	86.70%
[12]	10	373	Random Forest	84.19%
Our Proposed Model	22	464	AdaBoost	97.84%
			Bagging	98.28%
			Random Forest	99.35%

The comparison between AdaBoost, Bagging and Random Forest algorithms have been shown in Table V based on accuracy of each algorithm.

Table VI shows the comparison between our proposed model and several existing models. The existing models have also been developed using the algorithms we used in this study. So, we can compare the existing models with our model that we have proposed based on several factors such as, size and number of features of the datasets, algorithms and accuracy. The Table VI clarify that, our proposed model with AdaBoost, Bagging and Random Forest algorithms is better than the existing systems in terms of performance which have given the highest accuracy 97.84%, 98.28% and 99.35% respectively.

V. Conclusion

Though we have met some significant limitations, we have finished the study flourishingly with our expected outcomes. At the beginning stage, we have faced several problems for example, collection of real time data was one of the main problems and there were a number of missing information. But we have filled up the missing information by using Machine Learning techniques. Among the three algorithms we used, Random Forest gave the best performance than Bagging and AdaBoost, where Bagging performed better than AdaBoost. The highest accuracy is 99.35% has given by Random Forest. In future, we would like to run this study with different algorithms such as, Gradient Boosting, Neural Fuzzy Inference System and Meta Heuristic Search even with bigger dataset. We can develop an expert system with our study therefore, we can predict diabetes more productively and adequately. Also, we need to enhance our sample size **(464 instances)** and handle the missing data more efficient way.

References

[1] "9 Symptoms of Type 1 & Type 2 Diabetes: Complications, Causes & Diet", *MedicineNet*, 2019. [Online]. Available: https://www.medicinenet.com/diabetes_mellitus/article.htm. [Accessed: 05- Jul- 2019].

[2] "International Diabetes Federation - What is diabetes", *Idf.org*, 2019. [Online]. Available: https://www.idf.org/aboutdiabetes/what-is-diabetes.html. [Accessed: 08- Jul- 2019].

[3] "Statistics About Diabetes", *Diabetes.org*, 2019. [Online]. Available: https://www.diabetes.org/resources/statistics/statistics-about-diabetes. [Accessed: 10- Jul- 2019].

[4] R. Hira, M. Miah and D. Akash, "Prevalence of Type 2 Diabetes Mellitus in Rural Adults (¿_31years) in Bangladesh", *Faridpur Medical College Journal*, vol. 13, no. 1, pp. 20-23, 2018. [Accessed 16 July 2019].

[5] "Machine Learning - Definition and application examples", *Spotlightmetal.com*, 2019. [Online]. Available: https://www.spotlightmetal.com/machine-learning–definition-and-application-examples-a-746226/. [Accessed: 17- Jul- 2019].

[6] A. Mir and S. Dhage, "Diabetes Disease Prediction Using Machine Learning on Big Data of Healthcare", in *2018 Fourth International Conference on Computing Communication Control and Automation (ICCUBEA)*, Pune, India, 2018.

[7] D. Dutta, D. Paul and P. Ghosh, "Analysing Feature Importances for Diabetes Prediction using Machine Learning", in *2018 IEEE 9th Annual Information Technology, Electronics and Mobile Communication Conference (IEMCON)*, Vancouver, BC, Canada, 2018.

[8] M. Raihan, Muhammad Muinul Islam, Promila Ghosh, Shakil Ahmed Shaj, Mubtasim Rafid Chowdhury, Saikat Mondal, Arun More, "A Comprehensive Analysis on Risk Prediction of Acute Coronary Syndrome Using Machine Learning Approaches", in *2018 21st International Conference of Computer and Information Technology (ICCIT)*, Dhaka, Bangladesh, 2018, pp. 1 - 6.

[9] D. Kinge and S. Gaikwad, "Survey on data mining techniques for disease prediction", *International Research Journal of Engineering and Technology (IRJET)*, vol. 05, no. 01, pp. 630-636, 2018. [Accessed 21 July 2018].

[10] S. Manna, S. Maity, S. Munshi and M. Adhikari, "Diabetes Prediction Model Using Cloud Analytics", in *2018 International Conference on Advances in Computing, Communications and Informatics (ICACCI)*, Bangalore, India, 2018.

[11] D. Verma and N. Mishra, "Analysis and prediction of breast cancer and diabetes disease datasets using data mining classification techniques ", in 2017 International Conference on Intelligent Sustainable Systems (ICISS), Palladam, India, 2017.

[12] W. Xu, J. Zhang, Q. Zhang and X. Wei, "Risk prediction of type II diabetes based on random forest model", in *2017 Third International Conference on Advances in Electrical, Electronics, Information, Communication and Bio-Informatics (AEEICB)*, Chennai, India, 2017.

[13] H. Emblem, "When to use a Trimmed Mean", *Medium*, 2018. [Online]. Available: https://medium.com/@HollyEmblem/when-to-usea-trimmed-mean-fd6aab347e46. [Accessed: 24-Jul-2019].

[14] "TrimMean function R Documentation",*Rdocumentation.org*. [Online].Available:www.rdocumentation.org/packages/sscore/versions/1.44.0/topics/trimMean. [Accessed: 25-Jul-2019].

[15] "Median", *RDocumentation*, 2019. [Online]. Available:https://www.rdocumentation.org/packages/stats/versions/3.5.2/topics/median. [Accessed: 01- Aug- 2019].

[16] H. Kandan, "Bagging the skill of Bagging(Bootstrap aggregating).", *Medium*, 2018. [Online]. Available: https://medium.com/@harishkandan95/bagging-the-skill-of-bagging-bootstrap-aggregating-83c18dcabdf1. [Accessed: 04- Aug- 2019].

[17] Brownlee, J. (2016). *Master Machine Learning Algorithms*. 1st ed. pp.137-138.

[18] "How Random Forest Algorithm Works in Machine Learning", *Medium*, 2019. [Online]. Available: https://medium.com/@Synced/how-random-forest-algorithm-works-in-machine-learning-3c0fe15b6674. [Accessed: 06- Aug- 2019].

[19] "What is a False Positive Rate?", *Corvil*, 2019. [Online]. Available: https://www.corvil.com/kb/what-is-a-false-positive-rate. [Accessed: 07- Sep- 2019].

[20] I. Witten, E. Frank and M. Hall, Data Mining practical Machine Learning Tools and Techniques, 3rd ed. *Morgan Kaufmann*, 2011, pp. 166-580.