

**IMPLICATIONS OF META CLASSIFIERS FOR ONSET DIABETES
PREDICTION**

BY

**MD. ASHAF UDDAULA
ID: 161-15-7473
AND**

**MD. AL - AMIN HOSSAIN
ID: 161-15-7483
AND**

**MD. KHALID HOSSEN
ID: 161-15-7487**

This Report Presented in Partial Fulfillment of the Requirements for the
Degree of Bachelor of Science in Computer Science and Engineering

Supervised By

Ahmed Al Marouf
Lecturer
Department of CSE
Daffodil International University

Co-Supervised By

Shah Md. Tanvir Siddiquee
Assistant Professor
Department of CSE
Daffodil International University



**DAFFODIL INTERNATIONAL UNIVERSITY
DHAKA, BANGLADESH
JULY 2020**

APPROVAL

This Project titled “**Implications of Meta Classifiers for Onset Diabetes Prediction**”, submitted by **Md. Ashaf Uddaula, Md. Al - Amin Hossain** and **Md. Khalid Hossen** to the Department of Computer Science and Engineering, Daffodil International University, has been accepted as satisfactory for the partial fulfillment of the requirements for the degree of B.Sc. in Computer Science and Engineering and approved as to its style and contents. The presentation has been held on **09th July, 2020**.

BOARD OF EXAMINERS



Dr. Syed Akhter Hossain
Professor and Head

Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Chairman



Subhenur Latif
Assistant Professor

Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

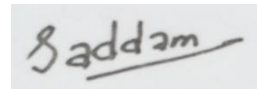
Internal Examiner



Raja Tariqul Hasan Tusher
Senior Lecturer

Department of Computer Science and Engineering
Faculty of Science & Information Technology
Daffodil International University

Internal Examiner



Dr. Md. Saddam Hossain
Assistant Professor

Department of Computer Science and Engineering
United International University

External Examiner

DECLARATION

We hereby declare that, this project has been done by us under the supervision of **Ahmed Al Marouf, Lecturer, Department of CSE** Daffodil International University. We also declare that neither this project nor any part of this project has been submitted elsewhere for award of any degree or diploma.

Supervised by:



Ahmed Al Marouf

Lecturer

Department of CSE

Daffodil International University

Co-Supervised by:



Shah Md. Tanvir Siddiquee

Assistant Professor

Department of CSE

Daffodil International University

Submitted by:

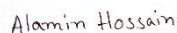


Md. Ashaf Uddaula

ID: 161-15-7473

Department of CSE

Daffodil International University

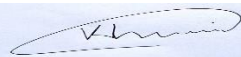


Md. Al - Amin Hossain

ID: 161-15-7483

Department of CSE

Daffodil International University



Md. Khalid Hossen

ID: 161-15-7487

Department of CSE

Daffodil International University

ACKNOWLEDGEMENT

First we express our heartiest thanks and gratefulness to almighty God for His divine blessing makes us possible to complete the final year project successfully.

We really grateful and wish our profound our indebtedness to **Ahmed Al Marouf, Lecturer**, Department of CSE Daffodil International University, Dhaka. Deep Knowledge & keen interest of our supervisor in the field of “*Data Mining*” to carry out this project. His endless patience ,scholarly guidance ,continual encouragement , constant and energetic supervision, constructive criticism , valuable advice ,reading many inferior draft and correcting them at all stage have made it possible to complete this project.

We would like to express our heartiest gratitude to **Prof. Dr. Syed Akhter Hossain, Head**, Department of CSE, for his kind help to finish our project and also to other faculty member and the staff of CSE department of Daffodil International University.

We would like to thank our entire course mate in Daffodil International University, who took part in this discuss while completing the course work.

Finally, we must acknowledge with due respect the constant support and patients of our parents.

ABSTRACT

In the data mining area, the prophecy of human diseases initiates a research zone for researchers by applying various machine learning algorithms with various patterns. As a modern community disease, diabetes is becoming one of the fastest-progressive human diseases in the world because of eating heavily sugared foods and lack of proper diet knowledge. In this era, most of the middle age people have confusion about the presence of diabetes in their bodies. That's why we choose to do research on diabetes. In this research, we scrutinized the classification performance of six Meta Classifiers named as Multiclass Classifier Updatable, Attribute Selected Classifier, Ada Boost M1, Logit Boost, Bagging, and Filtered Classifier for forecasting diabetes through cross-validation and percentage split techniques using in WEKA whereas as a diabetes dataset we used Pima Indians Database. And finally, according to win-rate from the Win-Draw-Loss table, the highest performance comes from Multiclass Classifier Updatable which has an 80% win-rate. On the other hand, in the measurement of highest individual accuracy, 81.9923% comes from both Attribute Selected Classifier and Filtered Classifier. According to the measurement of the highest average performance, 66% Split as a percentage split technique and Attribute Selected Classifier show the highest performance.

TABLE OF CONTENTS

CONTENT	PAGE NO
Board of Examiners	i
Declaration	ii
Acknowledgements	iii
Abstract	iv
 CHAPTER	
CHAPTER 01: INTRODUCTION	1-3
1.1 Diabetes	01
1.2 Data mining	01
1.3 Machine learning	02
1.4 Research Overview	03
 CHAPTER 02: RELATED WORKS	4-6
 CHAPTER 03: META CLASSIFIER OVERVIEW	7-8
3.1 Ada Boost M1	07
3.2 Bagging	07
3.3 Filtered Classifier	08
3.4 Logit Boost	08
3.5 Multiclass Classifier Updatable	08
3.6 Attribute Selected Classifier	08
 CHAPTER 04: EXPERIMENTAL MODEL	09-15
4.1 Dataset	09
4.2 Internal Classifier	10
4.2.1 Hoeffding Tree	10
4.2.2 REPTree	10

4.2.3 Random Forest	10
4.2.4 SGD	11
4.3 Performance Measured Used	11
4.3.1 Accuracy Rate of Classification	11
4.3.2 Precision	11
4.3.3 Recall	11
4.3.4 F-Score	12
4.3.5 Mean Absolute Error	12
4.3.6 Root Mean Squared Error	12
4.3.7 Matthews Correlation Coefficient	12
4.3.8 Kappa Statistic	13
4.4 Data Mining Techniques	13
4.4.1 Cross-Validation	13
4.4.2 Percentage Split	13
4.4.3 Win-Draw-Loss Table	13
4.4.4 Receiver Operating Characteristic Curve	14
4.4.5 Precision-Recall Curve	14
4.4.6 MCC Bar Chart	15
 CHAPTER 05: COMPARATIVE ANALYSIS	 16-22
 CHAPTER 06: CONCLUTION	 23
6.1 Summary	23
6.2 Future Implementation	23
 REFERENCES	 24

LIST OF FIGURES

FIGURES	PAGE NO
Figure 1.2.1: A View of Data Mining	02
Figure 5.1: Receiver Operating Characteristics (ROC) Curves of Meta Classifiers for Tested Positive Class according to 10-Fold Cross Validation	19
Figure 5.2: Receiver Operating Characteristics (ROC) Curves of Meta Classifiers for Tested Positive Class according to 66% Split	19
Figure 5.3: Precision-Recall Graph for 10-Fold Cross Validation	20
Figure 5.4: Precision-Recall Graph for 66% Split	20
Figure 5.5: Precision-Recall (PR) Curves of Meta Classifiers for Tested Positive Class according to 10-Fold Cross Validation	21
Figure 5.6: Precision-Recall (PR) Curves of Meta Classifiers for Tested Positive Class according to 66% Split	21
Figure 5.7: MCC Value for 10-Fold Cross Validation	22
Figure 5.8: MCC Value for 66% Split	22

LIST OF TABLES

TABLES	PAGE NO
Table 4.1.1: Attribute Description of Dataset [2]	9
Table 4.2.1: List of Internally used Classifiers in Meta Classifiers	10
Table 5.1: Cross-Validation wise performance metrics for Meta-Classifiers	16
Table 5.2: Kappa Statistic, MAE, RMSE, MCC & ROC Area of Meta Classifiers for Cross-Validation	17
Table 5.3: Percentage Split wise performance metrics for Meta-Classifiers	17
Table 5.4: Kappa Statistic, MAE, RMSE, MCC & ROC Area of Meta Classifiers for Percentage Split	18
Table 5.5: Win-Draw-Loss value for Meta Classifiers with Win Rate	18

CHAPTER 01

Introduction

1.1 Diabetes

Diabetes is a common chronic disease for our modern society. People getting used to diabetes day by day because of its availability in today's world. It can infect by genetically or, also by taking high sugared food. Nowadays it can be looked that, children are affected by diabetes increasingly due to the presence of diabetes to their parents. We know that we can't prevent diabetes but, we can control diabetes by controlling the sugar level in our blood circulation. But, most people don't know that they are affected by diabetes or, not yet. Even many people don't want to test the presence of diabetes because they don't want to lead their lives with a tight food schedule if they are affected. But, we need to checkup our body and maintain a disciplined food list in our life. Here, data mining helps people to check the presence of diabetes after analyzing some valid data.

1.2 Data mining

Data mining is a result of some periodic processes like data purifying, integration of data, selection of data, transformation of data, mining the data, evaluate pattern and lastly representation of knowledge that are helped to invent ultimate patterns, relationships, insights of enterprises measuring and managing where we are now and predicting where we will be in the tomorrow from huge data sets.

As a big asset for diabetes researchers, Data mining has performed a spontaneous role in diabetes research and also would be a beneficial way for our medical science. Actually, it builds a relationship with our medical healthcare resources. Data mining can identify clandestine knowledge from a large volume of diabetes-related data. Our belief is that data mining not only can significantly help in diabetes research but also it can ensure better quality health care for those patients who are affected by diabetes.

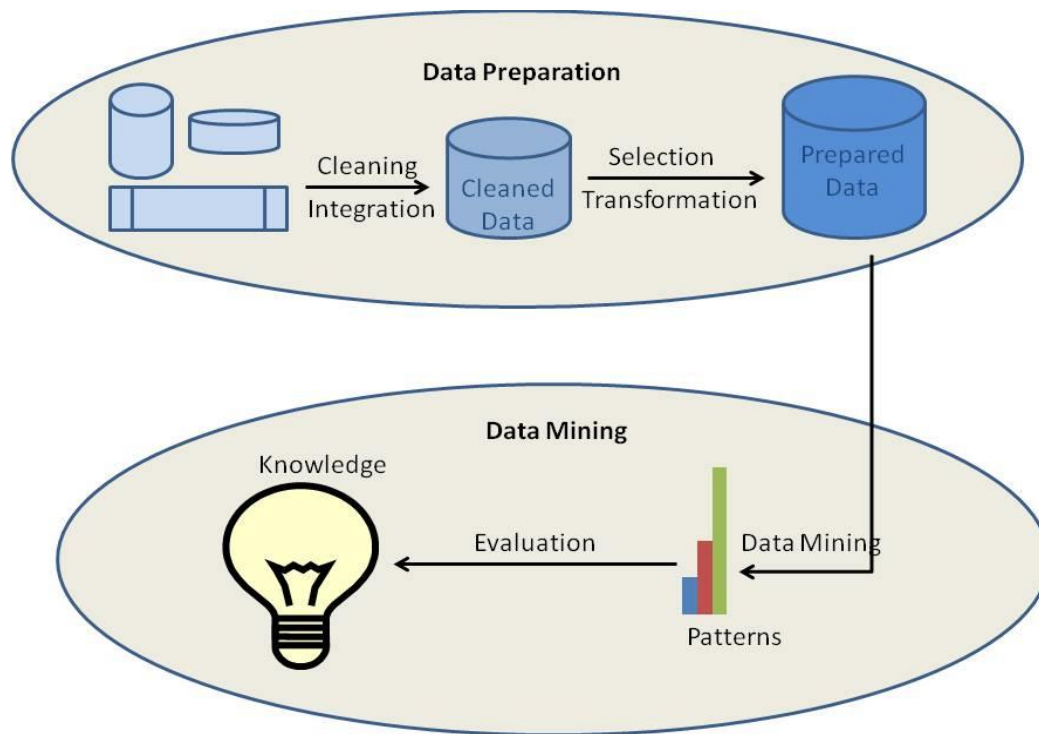


Figure 1.2.1: A View of Data Mining [12]

1.3 Machine learning

In diabetes research, data mining techniques are applied in some systematic ways. Machine learning algorithms used data mining techniques to build up the model and pattern to find out the accuracy rate of classification, prediction, relationship, and many others not only in the diabetes sector but also in diagnosis-related research sectors. Machine learning is an advanced study of mathematically proved algorithms and scientifically accepted statistical models that help computer-based hardware or, software systems to execute an appointed duty instead of using detail instructions, relying on models and hypothesis instead. It can be announced as a part of artificial intelligence. And machine learning algorithms set up a mathematical pattern based on specimen datum which is named as "training data", according to make prognostic or judgment instead of being in detail programmed to execute the task.

1.4 Research Overview

In Our research, we scrutinized the comparative view of six Meta classifier algorithms named as Multiclass Classifier Updatable, Attribute Selected Classifier, Ada Boost M1, Logit Boost, Bagging, and Filtered Classifier for forecasting diabetes through cross-validation and splitting techniques using in WEKA whereas as a diabetes dataset we used the most renowned Pima Indians Diabetes Database. We also estimated our comparative views with so many mathematically calculated tables especially the win-draw-loss table, many curves like ROC curves and Precision-Recall Curves and also many statistical graphs.

We found so many results from different comparative views. On the basis of the Win-Draw-Loss table, Multiclass Classifier Updatable has performed the highest performance with an 80% win-rate. On another comparative view, Attribute Selected Classifier and Filtered Classifier have given the highest accuracy 81.9923% individually. Then, if we talk about the averagely, then the percentage split technique named 66% Split and Attribute Selected Classifier accomplished the maximum performance comparing others where 66% Split done 80.08% and Attribute Selected Classifier done 80.69%.

Actually Meta Classifier is habitually a proxy to the principle classifier, used to supply excessive data preprocessing. That's why we choose Meta classifier for our comparative analysis and we hope that we could make a better comparison than others.

CHAPTER 02

Related Works

Diabetes has been announced globally as an epidemic. This pestilence diseases could be hugely attributed to the quick growth in the rate of physical inactivity, fatness, and overweight. According to the survey of WHO, about 350 million diabetes affected people are suffering in today's world. Diabetes will rank as seventh of the leading cause of death global by 2030. It is expected that diabetes will be rising by 50% during the upcoming 10 years. In low rated and middle-income countries, 4 out of 5 people are leading lives with diabetes [11].

Sandeep Kumar Budhani et al., [7] has studied three Meta Classifier Algorithms: Adaboost M1, Stacking, and Bagging which has applied to diabetes datasets: Hyperplane1 and Hyperplan2 and WEKA was used here as a data mining tool for measuring performance. According to this paper, Bagging has shown the highest accuracy rate for both Hyperplane1 and Hyperplan2 datasets and that are respectively 84.54% and 83.83%.

Lujain AlThunayan et al., [5] used a diabetes dataset for comparing Bayesian, Naive Bayes, J48, RandomForest, RandomTree, REP Tree, CART, and SMO classification algorithms that has helped to find out the best classification algorithm among them by measuring the accuracy of those classifiers.

Mirza Shuja et al., [9] wanted to present a detailed survey of various techniques of data mining that have been used to design prognostic models which will be helpful for other data mining researchers to predict diabetes.

P. Suresh Kumar et al., [11] proposed a pattern to reduce the problems created in most useful data mining techniques like classification and clustering. That helps to apply those techniques easily to collected diabetes data. Especially it helps to predict the risk of gestational diabetes.

Sajida Perveen et al., [13] made some models with a better classification's output of diabetes where the diabetes dataset is made with three age groups in the population of Canada, collected from Canadian Primary Care Sentinel Surveillance Network database. As a result, Adaboost has given better performance than bagging as well as J48 decision tree.

Lakshmi Devasena et al., [4] has scrutinized the proficiency of J48 Classifiers, Random Forest and REP Tree for the credit venture prognostics and compare their vigor through different measurements. As a dataset, the German Credit Dataset was used here. After final observation, Random Forest Classifier has performed best comparing respectively REP Tree Classifier and J48 Classifier. This paper didn't work with diabetes but, the scrutinized way in this paper was mind-blowing as comparing other's comparison works which will be helpful in diabetes research using data mining techniques.

Aiswarya Iyerv et al., [2] used Pima Indians Diabetes Dataset for analyzing the models using by Decision Tree and Naive Bayes classifiers, which help to build a most feasible model to search out the endemically and systematical dealing for diagnosing diabetes. This model will be helpful for flourishing the automation of diabetes scrutinize.

Razieh Asgarnezhad et al.,[10] has proposed a scheme with a proficient preprocessing technique together with absence value exploration(replace with mean) & optimize volition using the genetic algorithm on a diabetes dataset from Pima Indians database where this dataset has the lack of completeness. By using the SVM classifier which has predicted 84.35% accuracy rate which has the highest accuracy among the conferred comparison.

N. A. Nnamoko et al., [3] presented a diabetes prognostic model by investigating the way of predictions from different classification algorithms, repeating the task, could be utilized to output a greater performance comparing the highest separately learning algorithm. In this paper, RBF, RIPPER, SMO, Naïve Bayes, and C5.4 have trained to build up to five populated models. After comparison, a Meta model with a Logistic Regression algorithm was used to train and make final prognostics using the output of the

maximum and minimum performing algorithms as extra outputs. As a final result, C4.5 has performed the highest performance with a 77.9 % accuracy rate of classification and RBF has performed the lowest performance with a 73.6% accuracy rate of classification. On the other hand, the Meta Model accomplished 77.0% accuracy rate of classification.

Nithya Settu et al., [6] researched for diabetes and improved the performance of the filter algorithm by using Symmetrical Uncertainty Measure (SUM) and Novel Symmetrical Uncertainty Measure (NSUM) where SUM technique has reached 79.08% accuracy rate with 0.06 sec run time and NSU technique has achieved 89.12% accuracy rate in 0.03 sec run time. Both techniques had applied through WEKA.

CHAPTER 03

Meta Classifier Overview

Almost 20 Meta Classifiers are existing in WEKA. We choose only six of them for scrutinizing and comparing according to observing their accuracy rate in both cross-validation and splitting techniques. And they are Multiclass Classifier Updatable, Attribute Selected Classifier, Ada Boost M1, Logit Boost, Bagging, and Filtered Classifier.

3.1 Ada Boost M1

Ada Boost M1 is an extensively executed boosting algorithm that advantage to known well. For boosting a multiclass basis classifier as if the multiclass classification is consists of a problem, this classifier is used. Because of the too much weakness in the base classifier, Adaboost M1 won't work. But, after the interchange in Ada Boost M1 in one line only, it can be prepared as applicable. In our research, we used Hoeffding Tree as an internally used classifier.

3.2 Bagging

Bagging is also known as only Meta-Bagging also. Bagging is known as bootstrap aggregation. Bagging generates training data with bootstrap samples. It develops a distinct training set including numerous datasets. Various datasets are formed by unmethodical sampling happening with replacement. Each individual bootstrap specimen is used to train a classifier or, a regression function. Classification outputs are taken on the highest value of votes for classification intentions. For regression mean of prospective values are taken. Alternation is decreased and performance is developed for insecure classifiers that disagree meaningfully with tiny changes in the dataset. In the configuration of Bagging in our paper, REPTree is used as an internally used classifier.

3.3 Filtered Classifier

Restoring the architecture of the testing and training datum analogous to this classifier is used with different types of filters. Here in our paper, we configured Filtered Classifier where as a classifier we used Random Forest and as a filter we used Discretize.

3.4 Logit Boost

Logit Boost is the succession of the Ada Boost algorithm as it alternates the interpretative loss of the Ada Boost algorithm to temporary Bernoulli possibility loss. Logit Boost is used for the execution of preservative logistic regression. Here in our research work, we configured Logit Boost by determining Random Forest as internally used classifier.

3.5 Multiclass Classifier Updatable

Multiclass Classifier Updatable is an upgrade version of Multiclass Classifier. Error purification codes are modified with this classifier for achieving for much accuracy as this classifier is applied for categorizing events added to two classes. In our analysis, we used SGD as an internally used classifier in Multiclass Classifier Updatable.

3.6 Attribute Selected Classifier

The limit of the testing data and training data is reduced by Attribute Selected Classifier before being expired onto the classifier. Currently, researchers used base classifiers. So, the classifier is promoted several search ways are used during the stage of attribute selection. Here as an internally used classifier we used Hoeffding Tree, as evaluator we used cfsSubsetEval and as search, we used BFS.

CHAPTER 04

Experimental Model

We used 3 types of cross-validation techniques: 3-fold, 5-fold and 10-fold, and also 3-types of percentage split techniques: 66% Split, 75% Split and 80% Split. We collected our diabetes dataset from the Pima Indian Database and applied those techniques with declared Meta classifiers. Then, we scrutinized our output in a synchronized way. And finally, we presented our spontaneous opinion after some valid analyzations and comparisons.

4.1 Dataset

These declared data mining techniques have been applied to the Pima Indians Diabetes Database of National Institute of Diabetes and Digestive and Kidney Diseases. This dataset is collected from the UCI Machine Learning Repository. This dataset is consists of 768 record samples with 9 attributes which are used to predict the presence of diabetes after analyzing deeply. It is known as a renowned diabetes dataset in the data mining research area. A large number of diabetes-related data mining research has completed by using this data set. That's why we used this data set in our experiment. There is no doubt to accept its validity in the research area. Among 768 samples, 268 samples are tested positive and the rest of them are tested negative. All samples carried persons are Indian women with a minimum of 21 years old and live near Phoenix, Arizona, USA. The description of the attributes are listed in below.

TABLE 4.1.1: Attribute Description of Dataset [2]

Attribute	Description
Preg	Number of times pregnant
Plas	Plasma glucose concentration
Pres	Diastolic blood pressure (mm Hg)
Skin	Triceps skin fold thickness (mm)
Insu	2-Hour serum insulin(mu U/ml)
Mass	Body mass index (kg/m2)
Pedi	Diabetes pedigree function
Age	Patient Age (years)
Class	Class variable (0 or 1)

4.2 Internal Classifier

We have already declared in Meta Classifier Overview that in every Meta Classifier has to be used a main particular classifier with many various parameters which we said as an internally used classifier. Because, we know that Meta Classifier is habitually a proxy to the principle classifier, used to supply excessive data preprocessing. In our selected Meta Classifiers, we used the below classifiers as the internally used classifiers.

TABLE 4.2.1: List of Internally used Classifiers in Meta Classifiers

Meta Classifier's Names	Internally used Classifiers
Ada Boost M1	Hoeffding Tree
Bagging	REPTree
Filtered Classifier	Random Forest
Logit Boost	Random Forest
Multiclass Classifier Updatable	SGD
Attribute Selected Classifier	Hoeffding Tree

4.2.1 Hoeffding Tree

A Hoeffding tree known as a progressive decision tree which is able to learn from huge data drifts at any time with the assumption that the change according to time can't possible by distribution yield instances.

4.2.2 REPTree

REPTree is known as an algorithm of quick decision tree because it can build a decision tree using variability or, obtaining information. It can prune in a fast way as it also can deal with absence values using splitting parts into shreds. The missing values operation is also similar like C4.5 Algorithm. This pruning process is called reduced-error-pruning [1].

4.2.3 Random Forest

After so many combinations of tree predictors where every tree relies upon the values come from a random vector sampled automatically and with the equivalent dispensation

for the whole tree in the forest [4]. Pruning is not needed here. Because, until every node holds simply very small number of monitoring, trees can be generated.

4.2.4 SGD

SGD generally replaces all absence values and the nominal attributes transform in binary forms. The coefficients in the result are depending on the normalized data that's why SGD also normalizes whole attributes [8]. Actually it is a repetitive process for minimizing a goal task with proper blandness features.

4.3 Performance Measured Used

For measuring the performance of the declared Meta Classifier, we used numerous values that come from different sectors.

4.3.1 Accuracy Rate of Classification

Accuracy Rate of Classification is computed as exactly classified samples divided by the entire number of samples multiplied by 100. Exact classified sample is the sum of True-Positive (TP) and True-Negative.

$$\text{Accuracy Rate} = \frac{TP+TN}{Total} \times 100 \quad (1)$$

4.3.2 Precision

According to the Confusion Matrix, Precision is the ratio between true-positive samples and predicted yes samples.

$$\text{Precision} = \frac{TP}{TP+FP} \quad (2)$$

Here, TP+FP = Predicted Yes

4.3.3 Recall

Recall is also known as Sensitivity. According to the Confusion Matrix, Recall is the ratio true-positive samples and actual yes samples.

$$\text{Recall} = \frac{TP}{TP+FN} \quad (3)$$

Here, $TP+FN$ = Actual Yes

4.3.4 F-Score

F-Score is also called F1-Score or, F-Measure. The F-Score can give a more feasible measurement of a test implementation using both recall and precision. When the value of F-Score becomes 1 that indicates the perfection of both recall and precision.

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

4.3.5 Mean Absolute Error

MAE calculates the average measurement of the errors in a set of prognostics, except considering their way.

$$\text{MAE} = \frac{\sum_{i=1}^n |p_i - a_i|}{n} \quad (5)$$

4.3.6 Root Mean Squared Error

RMSE is the square root of the mean of squared differences between prophecy and actual espial.

$$\text{RMSE} = \sqrt{\frac{\sum_{i=1}^n (p_i - a_i)^2}{n}} \quad (6)$$

4.3.7 Matthews Correlation Coefficient

MCC calculates the quality of the classification which has two types. Actually, the value of MCC proposed the correlation coefficient among the predicted and noticed classification which is binary. According to Confusion Matrix, the formula of MCC will be,

$$MCC = \frac{TP \times TN - FP \times FN}{(TP + FP)(TP + FN)(TN + FP)(TN + FN)} \quad (7)$$

4.3.8 Kappa Statistic

Kappa Statistic is also known as Cohen's Kappa. Actually, it is used for quantifying the ability of reproduction of a distinct variable.

$$K = \frac{p_o - p_e}{1 - p_e} = 1 - \frac{1 - p_o}{1 - p_e} \quad (8)$$

Here, p_o = Observed Agreement and p_e = Expected Agreement.

4.4 Data Mining Techniques

We have already said that we used cross-validation and percentage split techniques in numerous ways. And scrutinizing the results in different ways as Precision-Recall Curve, ROC curve, bar chart using MCC values and Win-Draw-Loss table.

4.4.1 Cross-Validation

Cross-Validation is a heuristic works that arbitrarily classify the data into n-folds, each with nearly the similar number of records, makes n-models using the similar algorithms and training parameters where every model is trained with n-1 folds of the data and tested on the due fold, can be applied to search the best algorithm and its optimum training parameters.

4.4.2 Percentage Split

Percentage Split is a process of re-sampling that reserves n% of the rows as the training dataset for structuring the model and (n-100) % of the rows reserved as the test dataset to test the model. The target classifier is trained as opposed to the trained data. On the other hand, the classification accuracy is measured on the test dataset.

4.4.3 Win-Draw-Loss Table

The win-draw-loss table represents the winning rate comparing other classifiers. Here, the value of a win, draw or, a loss will be equal or, less than the total numbers of

comparison classifiers. Actually, it is the most applicable is biological research but, in the data mining research area as a comparison, it would be helpful. According to the value of the win, we can easily estimate the win-rate of a classifier.

$$\text{Win Rate} = \frac{\text{Total Number of Win}}{\text{Total Number of Comparison}} \times 100 \quad (9)$$

4.4.4 Receiver Operating Characteristic Curve

ROC curve is a graphical plot that representing the performance of a classifier at whole classification thresholds. According to confusion matrix, this curve consists of two parameters. One is called True-Positive Rate and another is called False-Positive Rate.

$$\text{TPR} = \frac{TP}{TP+FN} \quad (10)$$

$$\text{FPR} = \frac{FP}{FP+TN} \quad (11)$$

A classifier builds a model that has no skill if it illustrates at the point (0.5, 0.5) or, by a diagonal line that comes from the bottom left of the ROC Curve to the top right and contains an AUC of 0.5. A classifier builds a model has that has perfect skill if it illustrates by a line that comes from the bottom left of the ROC Curve to the top left of the ROC Curve and moved the top right of the ROC Curve.

4.4.5 Precision-Recall Curve

A precision-recall curve is a graphical plot for various thresholds where according to X-axis, the recall values are placed and according to Y-axis the precision values are placed. This curve is useful in applied data mining for estimating binary classification patterns. A classifier builds a model that has no skill if it illustrates by a diagonal line that comes from the top left (0, 1) of the precision-recall curve to the bottom right (1, 0). A skillful model is illustrated as a point at (1, 1). A classifier builds a model has that has perfect skill if it illustrates by a line that comes from the top left of the precision-recall curve to the top right of the precision-recall curve and moved to the bottom right of the precision-recall curve.

4.4.6 MCC Bar Chart

After plotting, the MCC values in the bar chart we can easily indicate the perfectly closed prediction because, if the value comes to close to +1 then, the result will be considered as close to perfect predictions. On the other hand, if the value comes to close to -1 then, it will be considered as close to worst predictions. And the value 0 indicates better than random prognostic.

CHAPTER 05

Comparative Analysis

For comparing, the below processes are followed in our study.

At first, we have applied numerous values with various internally used classifiers for both cross-validation and percentage split techniques. And we have selected some particular values and internally used classifiers (shown in TABLE 4.2.1) for applying again in the cross-validation and percentage split techniques on the basis of accuracy variance and differences from each other. Here, we have selected 3-fold, 5-fold, and 10-fold for cross-validation and also selected 66%, 75% and 80% for percentage split.

Then we have recorded the value for precision, recall, f-score, accuracy, kappa statistics, MAE, RMSE, MCC and ROC area from outputs which are helped us to enhance our comparison in a further step. In TABLE 5.1 and TABLE 5.3, we have added two extra parameters named average accuracy and standard deviation of Accuracy in both row and column with completing their calculation. All records have shown in TABLE 5.1, TABLE 5.2, TABLE 5.3 and TABLE 5.4.

TABLE 5.1: Cross-Validation wise performance metrics for Meta-Classifiers

Meta Classifiers	3-Fold Cross Validation				5-Fold Cross Validation				10-Fold Cross Validation				Average Accuracy	Standard Deviation Of Accuracy
	Precision	Recall	F-Score	Accuracy	Precision	Recall	F-Score	Accuracy	Precision	Recall	F-Score	Accuracy		
Ada Boost M1	0.764	0.770	0.763	76.9531%	0.758	0.763	0.758	76.3021%	0.763	0.768	0.764	76.8229%	76.69%	0.0034
Bagging	0.756	0.762	0.757	76.1719%	0.748	0.754	0.749	75.3906%	0.752	0.758	0.753	75.7813%	75.78%	0.0039
Filtered Classifier	0.748	0.749	0.748	74.8698%	0.708	0.710	0.709	70.9635%	0.724	0.730	0.726	73.0469%	72.96%	0.0195
Logit Boost	0.753	0.757	0.755	75.651%	0.739	0.742	0.740	74.2188%	0.744	0.749	0.746	74.8698%	74.91%	0.0072
Multiclass Classifier Updatable	0.764	0.770	0.763	76.9531%	0.757	0.763	0.756	76.3021%	0.776	0.780	0.771	77.9948%	77.08%	0.0085
Attribute Selected Classifier	0.758	0.764	0.756	76.4323%	0.753	0.759	0.751	75.9115%	0.771	0.776	0.770	77.6042%	76.65%	0.0087
Average	0.757	0.762	0.757	76.17%	0.743	0.748	0.743	74.85%	0.755	0.760167	0.755	76.02%		
Standard Deviation	0.0063	0.0081	0.0056	0.0081	0.0189	0.0204	0.0182	0.0206	0.0192	0.0187	0.0172	0.0186		

TABLE 5.2: Kappa Statistic, MAE, RMSE, MCC & ROC Area of Meta Classifiers for Cross-Validation

Meta Classifier	3-Fold Cross Validation					5-Fold Cross Validation					10--Fold Cross Validation				
	Kappa Statistic	MAE	RMSE	MCC	ROC Area	Kappa Statistic	MAE	RMSE	MCC	ROC Area	Kappa Statistic	MAE	RMSE	MCC	ROC Area
Ada Boost M1	0.4703	0.3023	0.408	0.475	0.815	0.4597	0.308	0.4161	0.463	0.803	0.4735	0.308	0.415	0.476	0.796
Bagging	0.4576	0.307	0.3997	0.460	0.827	0.4405	0.3105	0.4039	0.443	0.820	0.4498	0.315	0.4063	0.452	0.812
Filtered Classifier	0.4445	0.3179	0.4233	0.445	0.784	0.3582	0.3113	0.4303	0.358	0.789	0.3894	0.3083	0.4283	0.391	0.786
Logit Boost	0.4561	0.2505	0.4554	0.457	0.811	0.4236	0.2596	0.4593	0.424	0.809	0.4357	0.258	0.4585	0.437	0.808
Multiclass Classifier Updatable	0.4693	0.2305	0.4801	0.475	0.724	0.4529	0.237	0.4868	0.459	0.716	0.4868	0.2201	0.4691	0.497	0.730
Attribute Selected Classifier	0.4524	0.3007	0.4135	0.460	0.807	0.4423	0.303	0.4058	0.449	0.821	0.4839	0.2987	0.4046	0.490	0.820

After scrutinizing TABLE 5.1 and TABLE 5.3, Attribute Selected Classifier and Filtered Classifier both have given the highest accuracy of 81.9923% individually in the 66% split (TABLE 5.3). And on average, 66% split has achieved the highest accuracy of 80.08% as a technique and Attribute Selected Classifier has achieved the highest accuracy of 80.69% as a classifier (TABLE 5.3). Individually in TABLE 5.1, as a classifier, Multiclass Classifier Updatable has reached the highest accuracy of 77.9948% in the 10-fold cross-validation and on average, 3-fold cross-validation has got the highest accuracy of 76.17%.

TABLE 5.3: Percentage Split wise performance metrics for Meta-Classifiers

Meta Classifier	66% Split				75% Split				80% Split				Average Accuracy	Standard Deviation Of Accuracy
	Precision	Recall	F-Score	Accuracy	Precision	Recall	F-Score	Accuracy	Precision	Recall	F-Score	Accuracy		
Ada Boost M1	0.798	0.801	0.799	80.0766%	0.794	0.792	0.793	79.1667%	0.779	0.779	0.779	77.9221%	79.06%	0.0108
Bagging	0.770	0.778	0.771	77.7778%	0.792	0.797	0.793	79.6875%	0.775	0.779	0.777	77.9221%	78.46%	0.0106
Filtered Classifier	0.816	0.820	0.814	81.9923%	0.771	0.776	0.760	77.6042%	0.779	0.786	0.776	78.5714%	79.39%	0.0230
Logit Boost	0.771	0.778	0.773	77.7778%	0.798	0.802	0.799	80.2083%	0.786	0.792	0.787	79.2208%	79.07%	0.0122
Multiclass Classifier Updatable	0.804	0.808	0.804	80.8429%	0.808	0.813	0.808	81.25%	0.788	0.792	0.790	79.2208%	80.44%	0.0107
Attribute Selected Classifier	0.816	0.820	0.815	81.9923%	0.799	0.802	0.800	80.2083%	0.793	0.799	0.794	79.8701%	80.69%	0.0114
Average	0.796	0.801	0.796	80.08%	0.794	0.797	0.792	79.69%	0.783	0.788	0.784	78.79%		
Standard Deviation	0.0208	0.0191	0.0195	0.0192	0.0124	0.0124	0.0167	0.0123	0.0068	0.0080	0.0075	0.0079		

TABLE 5.4: Kappa Statistic, MAE, RMSE, MCC & ROC Area of Meta Classifiers for Percentage Split

Meta Classifier	66% Split					75% Split					80% Split				
	Kappa Statistic	MAE	RMSE	MCC	ROC Area	Kappa Statistic	MAE	RMSE	MCC	ROC Area	Kappa Statistic	MAE	RMSE	MCC	ROC Area
Ada Boost M1	0.5347	0.323	0.3939	0.535	0.788	0.5276	0.2952	0.3912	0.528	0.838	0.4912	0.2983	0.3903	0.491	0.816
Bagging	0.4599	0.2966	0.3838	0.465	0.841	0.5214	0.2936	0.3783	0.523	0.846	0.4798	0.2981	0.3848	0.481	0.834
Filtered Classifier	0.5608	0.289	0.3755	0.568	0.847	0.4328	0.2861	0.3872	0.454	0.835	0.4685	0.2791	0.3844	0.479	0.831
Logit Boost	0.4671	0.2365	0.4304	0.470	0.831	0.5356	0.2275	0.4162	0.537	0.838	0.4993	0.2297	0.4236	0.503	0.826
Multiclass Classifier Updatable	0.5406	0.1916	0.4377	0.544	0.760	0.5523	0.1875	0.433	0.557	0.765	0.5104	0.2078	0.4558	0.511	0.750
Attribute Selected Classifier	0.5667	0.2616	0.3719	0.571	0.854	0.5396	0.2667	0.3922	0.540	0.832	0.5177	0.2609	0.3887	0.521	0.803

TABLE 5.5: Win-Draw-Loss value for Meta Classifiers with Win Rate

Meta Classifier	Ada Boost M1	Bagging	Filtered Classifier	Logit Boost	Multiclass Classifier Updatable	Attribute Selected Classifier	Win Rate
Ada Boost M1	-----	4-1-1	4-0-2	4-0-2	0-2-4	2-0-4	46.67%
Bagging	1-1-4	-----	4-0-2	3-1-2	0-0-6	0-0-6	26.67%
Filtered Classifier	2-0-4	2-0-4	-----	1-0-5	1-0-5	0-1-5	20%
Logit Boost	2-0-4	2-1-3	5-0-1	-----	0-1-5	0-1-5	30%
Multiclass Classifier Updatable	4-2-0	6-0-0	5-0-1	5-1-0	-----	4-0-2	80%
Attribute Selected Classifier	4-0-2	6-0-0	5-1-0	5-1-0	2-0-4	-----	73.33%

Here, we have produced a win-draw-loss shown in TABLE 5.5. We have calculated the win rate for each Meta Classifier also shown in TABLE 5.5. According to TABLE 5.5, we have identified that Multiclass Classifier Updatable has achieved the highest win rate.

For the view of comparison, from ROC area values of TABLE 5.2 and TABLE 5.4, we have made ROC curves for each selected Meta Classifier according to 10-fold cross-validation (Figure 5.1) and 66% split (Figure 5.2).

Receiver Operating Characteristics (ROC) Curves of Meta Classifiers for Tested Positive Class according to 10-Fold Cross Validation

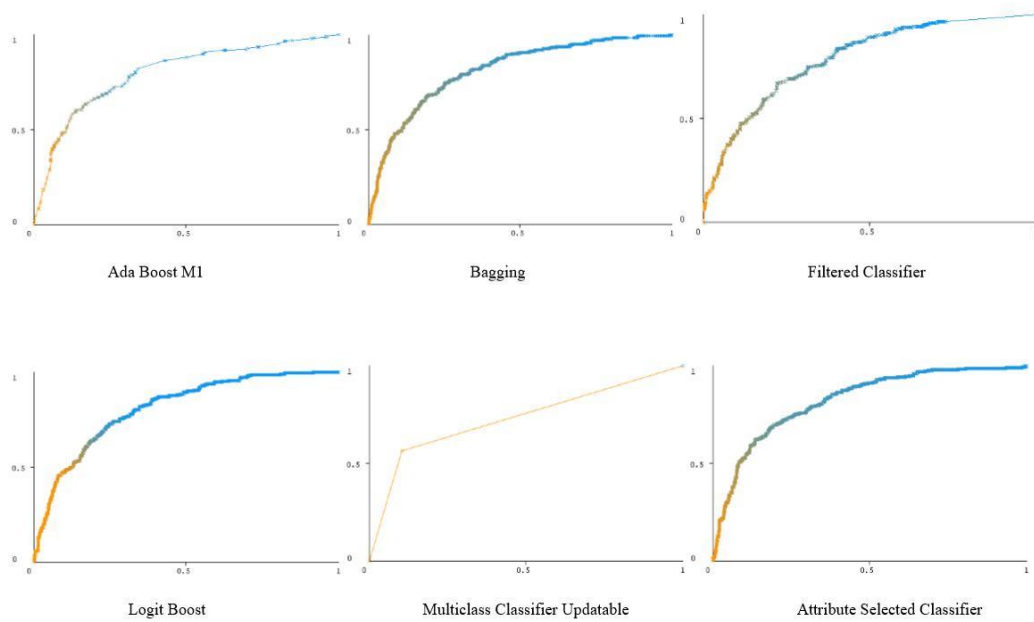


Figure 5.1: Receiver Operating Characteristics (ROC) Curves of Meta Classifiers for Tested Positive Class according to 10-Fold Cross Validation

Receiver Operating Characteristics (ROC) Curves of Meta Classifiers for Tested Positive Class according to 66% Split

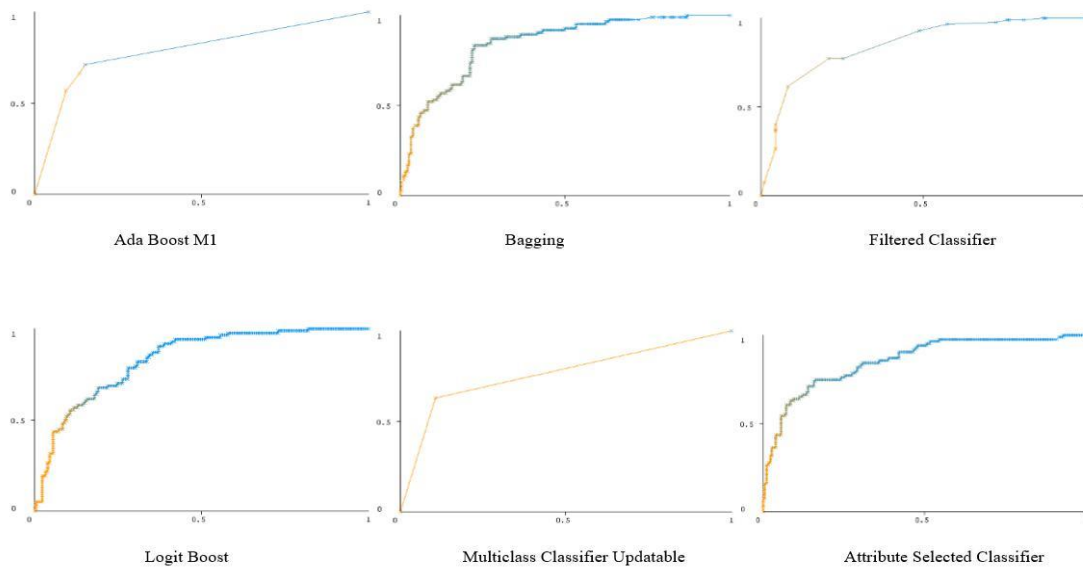


Figure 5.2: Receiver Operating Characteristics (ROC) Curves of Meta Classifiers for Tested Positive Class according to 66% Split

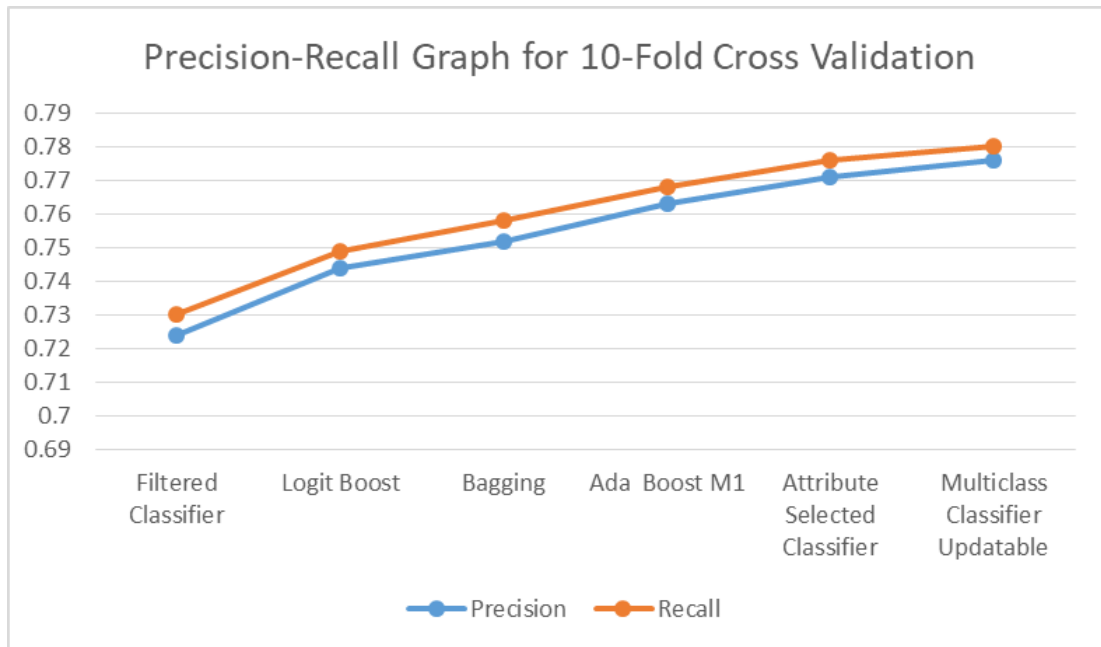


Figure 5.3: Precision-Recall Graph for 10-Fold Cross Validation

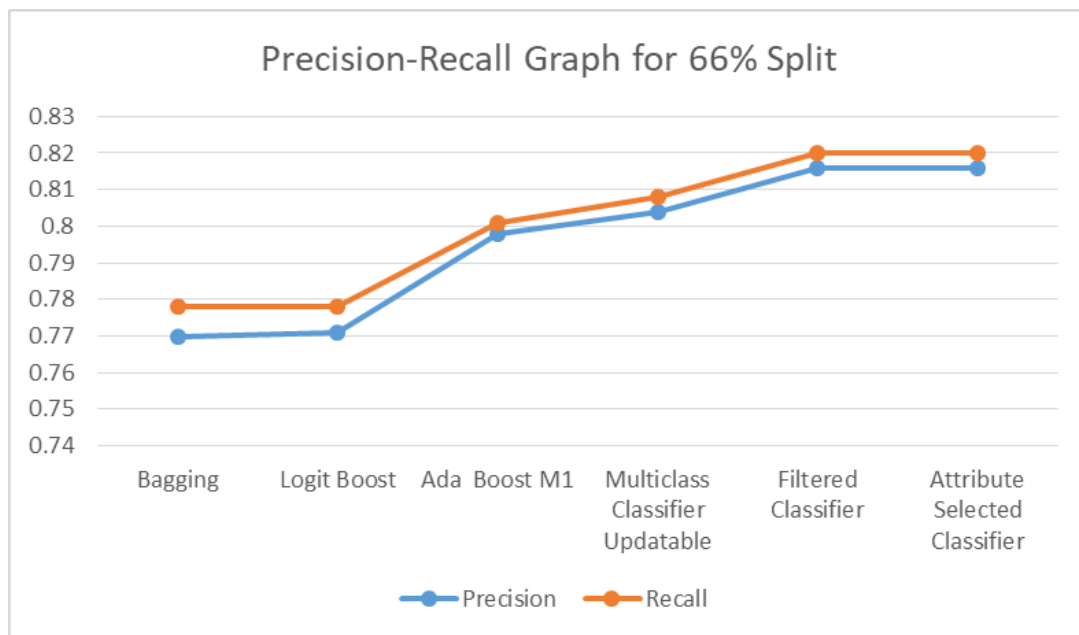


Figure 5.4: Precision-Recall Graph for 66% Split

For comparing another angle of view, from precision and recall values of TABLE 5.1 and TABLE 5.3, we have made precision-recall graphs for both 10-fold cross-validation (Figure 5.3) and 66% split (Figure 5.4) and also have shown the precision-recall curves (Figure 5.5 & Figure 5.6) for every declared Meta Classifier.

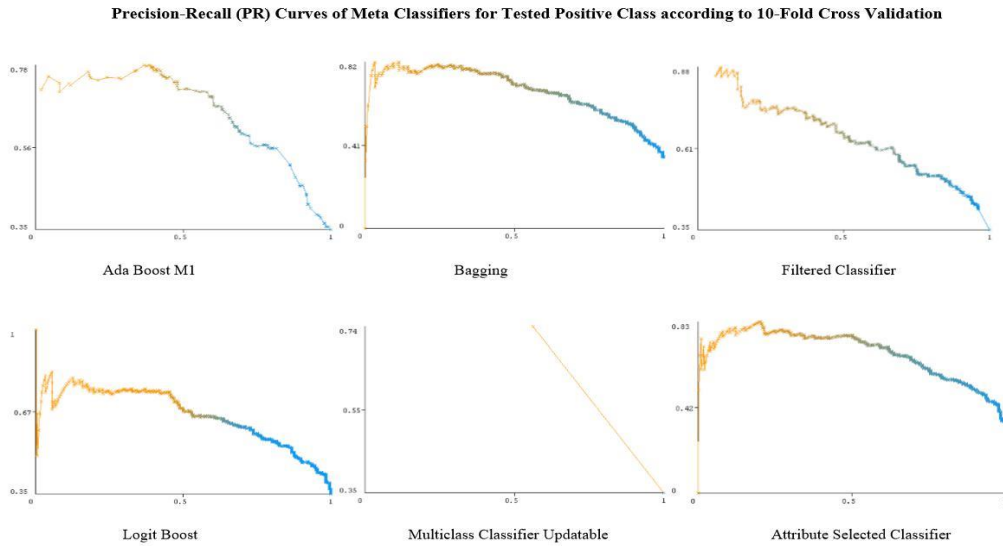


Figure 5.5: Precision-Recall (PR) Curves of Meta Classifiers for Tested Positive Class according to 10-Fold Cross Validation

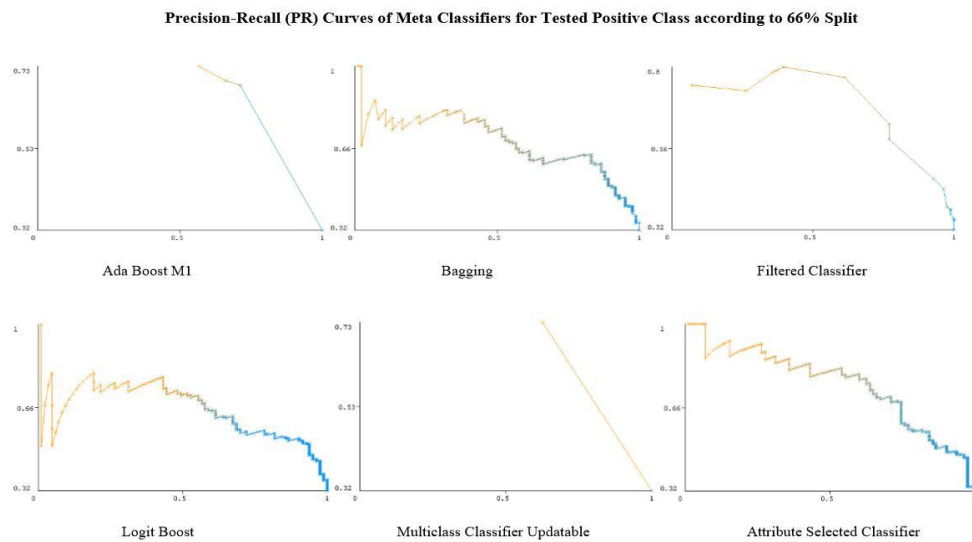


Figure 5.6: Precision-Recall (PR) Curves of Meta Classifiers for Tested Positive Class according to 66% Split

Lastly, we have represented two bar charts (Figure 5.7 & Figure 5.8) from MCC values of TABLE 5.2 and TABLE 5.4 for calculating the qualities of prediction.

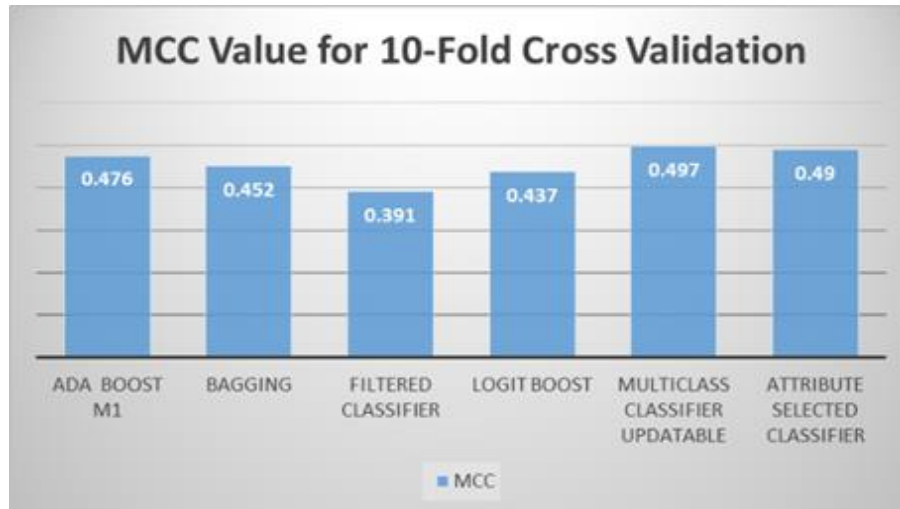


Figure 5.7: MCC Value for 10-Fold Cross Validation

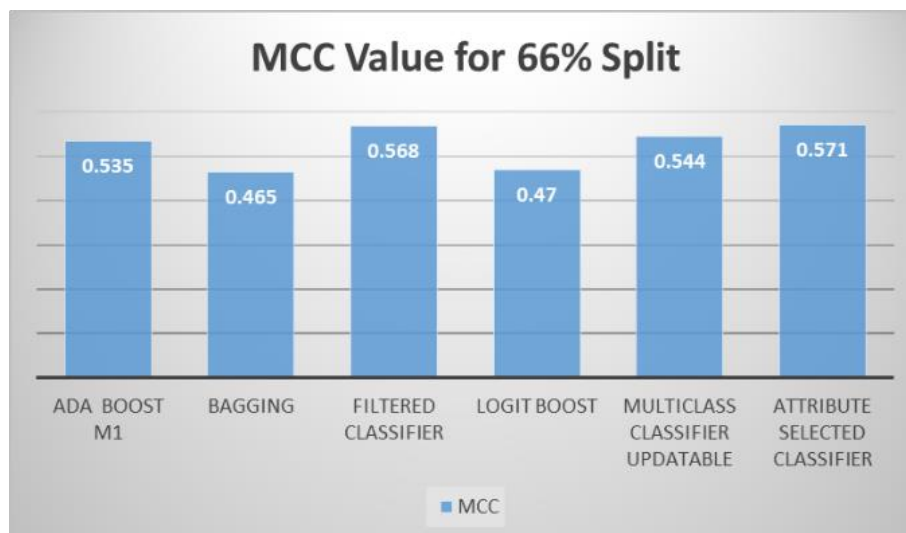


Figure 5.8: MCC Value for 66% Split

CHAPTER 06

Conclusion

6.1 Summary

This comparison investigated the overall efficiency of the six Meta Classifiers namely, Multiclass Classifier Updatable, Attribute Selected Classifier, Ada Boost M1, Logit Boost, Bagging, Filtered Classifier for forecasting diabetes. And finally, Attribute Selected Classifier and Filtered Classifier perform better than others in the individual platform. On average, 66% split as a technique and Attribute Selected Classifier as a classifier give the best performance. But, most importantly according to the win rate, Multiclass Classifier Updatable takes place over all of them.

6.2 Future Implementation

In the future, we will be focused on the better accuracy rate of classification and also need to find out the better data mining technique applying different machine learning algorithms. For that, we need to be tested our data set in different ways. We decided to use more renowned diabetes datasets for further research. We also decided that we will not only apply our research techniques in the diabetes sector but also we would like to do our research on other medical sectors.

REFERENCES

- [1] Witten, Ian H., Eibe Frank, Mark A. Hall, and Christopher J. Pal. Data Mining: Practical machine learning tools and techniques. Morgan Kaufmann, 2016.
- [2] Iyer, Aiswarya, S. Jeyalatha, and Ronak Sumbaly. "Diagnosis of diabetes using classification mining techniques." arXiv preprint arXiv:1502.03774 (2015).
- [3] Nnamoko, Nonso Alex, Farath N. Arshad, David England, and Jiten Vora. "Meta-classification Model for Diabetes onset forecast: a proof of concept." In 2014 IEEE International Conference on Bioinformatics and Biomedicine (BIBM), pp. 50-56. IEEE, 2014.
- [4] Devasena, C. Lakshmi. "Comparative analysis of random forest rep tree and j48 classifiers for credit risk prediction." In International Conference on Communication, Computing and Information Technology (ICCCMIT-2014). 2014.
- [5] Settu, Nithya, and M. Rajasekhara Babu. "Enhancing the Performance of Decision Tree Using NSUM Technique for Diabetes Patients." In Internet of Things and Personalized Healthcare Systems, pp. 13-20. Springer, Singapore, 2019.
- [6] Budhani, Sandeep Kumar, C. Jha, and Amir Ahmad. "Comparative Study of Meta Classification Algorithm: Bagging, AdaboostM1 and Stacking with Concept Drift based Synthetic Dataset Hyperplane1 and Hyperplane2." International Journal of Engineering Science 15927 (2018).
- [7] "SGD", Weka.sourceforge.net, 2019. [Online]. Available: <http://weka.sourceforge.net/doc.dev/weka/classifiers/functions/SGD.html>. [Accessed: 11- Nov- 2019].
- [8] Shuja, Mirza, Sonu Mittal, and Majid Zaman. "Diabetes Mellitus and Data Mining Techniques: A survey." (2019).
- [9] Asgarnezhad, Razieh, Maryam Shekofteh, and FARSAD ZAMANI BOROUJENI. "Improving Diagnosis of Diabetes Mellitus Using Combination of Preprocessing Techniques." Journal of Theoretical & Applied Information Technology 95, no. 13 (2017).
- [10] Kumar, P. Suresh, and V. Umatejaswi. "Diagnosing diabetes using data mining techniques." International Journal of Scientific and Research Publications 7, no. 6 (2017): 705-709.
- [11] "04 - Data Mining Processes", Wideskills.com, 2019. [Online]. Available: <https://www.wideskills.com/sites/default/files/subjects/Data%20Mining%20Tutorial/04/image1.jpeg>. [Accessed: 09- Nov- 2019].
- [12] Perveen, Sajida, Muhammad Shahbaz, Aziz Guergachi, and Karim Keshavjee. "Performance analysis of data mining classification techniques to predict diabetes." Procedia Computer Science 82 (2016): 115-121.

Diabetes Detection v2

ORIGINALITY REPORT

30%

SIMILARITY INDEX

5%

INTERNET SOURCES

22%

PUBLICATIONS

3%

STUDENT PAPERS

PRIMARY SOURCES

1

"Implications of Meta Classifiers for Onset Diabetes Prediction", International Journal of Innovative Technology and Exploring Engineering, 2020

Publication

22%

2

bmcbioinformatics.biomedcentral.com

Internet Source

<1%

3

www.ijitee.org

Internet Source

<1%

4

Huiqing Wang, Jingjing Wang, Chunlin Dong, Yuanyuan Lian, Dan Liu, Zhiliang Yan. "A Novel Approach for Drug-Target Interactions Prediction Based on Multimodal Deep Autoencoder", Frontiers in Pharmacology, 2020

Publication

<1%

5

Submitted to Victorian Institute of Technology

Student Paper

<1%

6

"Intelligent Computing, Networking, and Informatics", Springer Science and Business Media LLC, 2014

<1%