SGBBA: An Efficient Method for Prediction System in Machine Learning using Imbalance Dataset

Saiful Islam[1]
Computer Science & Engineering
Chittagong University of Engineering & Technology
International Islamic University of Chittagong
Chattogram
Bangladesh

Umme Sara[2], Anichur Rahman[4]
Dipanjali Kundu[5], Mahedi Hasan[8]
Department of Computer Science and Engineering
National Institute of Textile Engineering and Research
(NITER), Dhaka
Bangladesh

Abu Kawsar[3]
Department of Information and Communication
Technology, Government Maulana Mohammad Ali College
Tangail, Bangladesh

Diganta Das Dipta[6]
Department of Computer Science and Engineering,
Chittagong University of Engineering & Technology
Chattogram, Bangladesh

A.N.M. Rezaul Karim[7]
Department of Computer Science and Engineering
International Islamic University of Chittagong
Chattogram,Bangladesh

***Abstract*—A real world big dataset with disproportionate classification is called imbalance dataset which badly impacts the predictive result of machine learning classification algorithms. Most of the datasets faces the class imbalance problem in machine learning. Most of the algorithms in machine learning work perfectly with about equal samples counts for every class. A variety of solutions have been suggested in the past time by the different researchers and applied to deal with the imbalance dataset. The performance of these methods is lower than the satisfactory level. It is very difficult to design an efficient method using machine learning algorithms without making the imbalance dataset to balance dataset. In this paper we have designed an method named SGBBA: an efficient method for prediction system in machine learning using Imbalance dataset. The method that is addressed in this paper increases the performance to the maximum in terms of accuracy and confusion matrix. The proposed method is consisted of two modules such as designing the method and method based prediction. The experiments with two benchmark datasets and one highly imbalanced credit card datasets are performed and the performances are compared with the performance of SMOTE resampling method. F-score, specificity, precision and recall are used as the evaluation matrices to test the performance of the proposed method in terms of any kind of imbalance dataset. According to the comparison of the result of the proposed method computationally attains the effective and robust performance than the existing methods.**

Keywords—Imbalanced dataset; sub sample; accuracy; fraud; confusion matrix; bagging

I. INTRODUCTION

Now-a-days imbalanced classification from the two-class imbalance dataset pose a severe problem of data science and machine learning where every class has supremacy over another class. The dominant class with more data samples is called the majority class and the other class which has fewer samples is called the minority class. This question makes the machine learning models and algorithms more skewed towards the majority class ignoring the minority class where the minority class is more relevant. In such situation, it is notably important that we should develop a method for both majority and minority class dataset without discrimination to either of the majority and minority class. For most machine learning algorithms, it is very critical to identify rare objects than common objects [27, 28]. Data mining using imbalance dataset can be used in various practical fields such as direct marketing [30], software quality prediction [29], multi object genetic sampling [1] and rare event detection such as human decision making response [2]. This is a critical factor invoked for many practical uses, such as the detection of credit card fraud, disease prediction, market share prediction etc. Without considering imbalance problem in dataset the prediction result of newly developed model and algorithm are overwhelmed through majority classification and left out via minority class. Samples of minority classes are misclassified than samples of the dominant class. Credit card fraud detection is a real-world class imbalanced problem where non frauds 99.83% and frauds 0.17% of the dataset. In this regards, the level of fraud elegance is lower than the level of non-fraud magnificence. It is in this situation that kind 1 error fee is befallen at some stage in the prediction. It means that the non-fraud is graded wrongly over the fraud. It is most important that the machine learning model should be developed properly so that the imbalance problem no more exists in the dataset. If it is failed then the model provides more accuracy that is meaningless in the data science due to result from meaningless matric. Hence this higher accuracy is no longer reliable and realistic for model performance.

II. Background Study

A variety of processes have been proposed with the aid of the researchers to resolve the imbalance dataset hassle in the machine learning getting to know. These approaches belong to the following level of solutions such as algorithms level, data level, cost sensitive and ensemble solutions. The data level solution is the most popular and widely used method that is data preprocessing based solution. Data preprocessing is performed by resampling the imbalance dataset such as oversampling or super sampling the class with minorities [3], undersampling of class with majority [4] and combining the oversampling and undersampling through bagging [8] and boosting [7] methods such as SMOTEBoost [9], RUSBoost [10], Overbagging [11], Underbagging [12]. Both oversampling and undersampling methods creates various limitations in the dataset that make the prediction result and performance unreliable.

Japkowics et al. [18] explained the effect of imbalanced dataset very nicely in the machine learning classification algorithms. She experimented and presented three different strategies such as under-sampling, resampling and recognition based scheme. The random resampling refers to the oversampling the minority class that has a bit number of samples than the majority class at random until the wide variety of samples of the minority class is matched with the majority class. Random undersampling refers to the elimination of samples from the majority class which has an enormous number of samples than the minority class until the number of majority class samples equals the number of minority class samples.

Japkowics et al. [19] combined methods of oversampling and undersampling, called hybrid method. They introduced that the test examples are graded by a measure of trust and the lift is used as the assessment criterion. In the first experiment, they oversampled the smaller samples and in the second experiment, they undersampled the greater samples. Their aggregation of oversampling and undersampling did now not offer any significant improvement of performance in the lift of indexing. The oversampling method increases the possibility of data redundancy, depending on how instances are generated. For removing this problem, few approaches have been introduced, such as the Modified Synthetic Minority Oversampling Technique (MSMOTE) [21], and Adaptive Synthetic Sampling (ADASYN0) [22]. Another concern is that the instance replication appears to increase the computational cost of the learning process [23]. By comparison, random undersampling (RUS) is a method that shrinks the majority class though it is easy to use. As a result it may however delete some useful data from the dataset. To solve this percussion, the One Sided-Selection (OSS) technique [24] is used that cuts out the redundancy, noise that close to the boundary instances from the majority class. Border instances are discovered by using Tomek links and instances. In the clustering based under sampling [3] method the dataset is split into two classes as a form of majority and minority. Then clustering based undersampling is applied to eliminate the few samples of majority class data. After that, the reduced majority class data set will then be combined with the minority class dataset to form a balanced dataset. The classifier is finally trained using the balanced dataset. The problem with this approach is that certain essential data samples are omitted from the original dataset which may make the end result less accurate. In the Repeated random sub sampling [4] methods a number of samples from the original dataset are chosen and then the samples are divided into a number of sub-samples with the same number of instances in each class. After that, every sub-sample is fitted by the classification algorithm. Finally, the results are determined by majority vote on all sub-samples. The problem with this model is that the entire data set is not used in the experiment which may result in the final prediction being less accurate. SMOTE (Synthetic minority oversampling technique) [13] is one of the data science approaches that is most used and famous in the data science and machine learning where a synthetic minority class training instances are generated by spontaneously selected data instances based on interpolation with minority class. The SMOTE identifies each instance's k-nearest (typically k=5) neighbors from the minority class and then creates new instances synthetically as a convex combination that connects the two instances of the feature space to its k-nearest neighbors. Galar et al. [19], SMOTEBoost [20] is one of the most commonly used and popular methods that combines Synthetic Minority Oversampling Technique (SMOTE) and a rule-based standard boosting procedure where all instances that are misclassified are given equal weights. The SMOTEBoost synthetically generates instances of a rare or minority class to indirectly change the weights of a skewed minority class distribution, which reduces the variance. Consequently, removing the data samples from the original dataset can result in inaccurate prediction.

RUS Boosting (Random undersampling): In this RUS some data samples are randomly removed from the majority class of the dataset before the boosting procedure. Seitfort et al. [15] proposed a RUSBoosting approach that combines random under sampling method with a boosting procedure providing an effective and efficient method for improving classification performance when the dataset is imbalanced. It presents simple, effective, efficient, faster, easy alternative solution of SMOTEBoost for learning from disproportional dataset in machine learning. Under Bagging: Recently combining multiple classifiers into one classifier as ensembles classifiers has become more popular and considered as more promising approach in machine learning classification. The UnderBagging is essentially a combination of a random sampling technique and a bagging method. In Barandela et al. [16], first uses UnderBagging approach where majority or dominant class instances were sampled and then a balanced training data set was used construct a K nearest(typically K=1) neighbor Ensemble classifier based on bagging. Galar et al. [17] suggested a hybrid approach using a variety of balanced training sets to train classifier ensembles where each balanced training dataset was used for a single classifier. Then, a number of classifiers were then merged into one ensemble classifier by a hybrid bagging approach to achieve higher output while more classifiers made it more complex. The approaches to enhancing the classifier's overall accuracy are called the algorithmic level solution. There are two algorithmic level solutions, including the known recognized and sensitive solution. The SVM one-sided class method [25] is a known technique that takes into account only one class during the learning process. The support vector model in single-class SVM is trained on data that can only be trained by one normal class. A dynamic sampling method (DyS) for multilayer perceptron (MLP) [26] is a sensitive based approach where the probability of the selected sample is estimated by feeding the every sample to the current MLP.

The limitations are as:

- The oversampling method produces duplicate data sample in the dataset that may affect the overall prediction performance.
- Although under sampling is better than the oversampling, it removes important data sample from the dataset.
- Data redundancy and data hiding.

To remove above mentioned limitations of the existing methods we propose a novel predicting algorithm–The sub group based blanching method solutions of the imbalanced dataset that maximizes the effectiveness of the predictive result.

The contribution of this research can be summarized as follows:

- Firstly, we present a machine learning based prediction algorithm for dealing with the imbalance dataset that separates the data set into two groups i.e. majority class based dataset and minority class dataset. Then a balance dataset is made by taking the equal number of samples of minority class based dataset from the majority class based dataset with the samples of the minority class based dataset.
- Finally, we conduct *experiments* to evaluate the effectiveness of our proposed machine learning based prediction method in terms of imbalance dataset. The experimental results show that our proposed method significantly outperforms than the existing methods according to various test cases.

The remainder of the paper is organized as the following sections. In Section III, the suggested method is presented. Section IV outlines the assessment and experimental findings of the proposed method. Finally, Section V concludes the research study.

III. Proposed Methodology

The overall dataset is divided into two sub datasets as a dataset of minority class and majority class. The majority class dataset is then split into a number of sub-datasets equal to the total number of minority class data samples. Now the minority class dataset is combined with each sub-dataset of the majority class to create a balanced dataset before the sub-datasets of the majority class are used only with a single minority class sub-dataset. Once the minority class dataset is combined with a majority class sub dataset, the prediction model is tested and applied with the combined balanced dataset and the result is added as a grand total result. After all implementation of all sub-sample balanced data sets has been completed, the grand total result is averaged by the total number of sub-sample balance dataset. Eliminating all issues with current methods, such as deleting and duplicating essential data samples from the initial dataset, the proposed method does better than other existing methods.

Suppose, the dataset includes N samples. The N samples are divided into N_{max} and N_{min}, where N_{max} is the total number of samples in the majority class and N_{min} is the total number of samples in the minority class i.e. N=N_{max} + N_{min}. The N_{max} is divided into $N_{\max\, i}$ sub samples as equivalent to the N_{min} where i=1, 2, 3... N_{min}. Now each group of $N_{\max\, i}$ samples is merged to the N_{min} samples as a balanced data set. Such as the balanced dataset = $Merge(N_{\max\, i}, N_{min})$ where $N_{\max\, i}$ is a group samples of the majority class data and N_{min} is the group of samples of the minority class data. Finally, every balanced dataset produced is applied to the classification techniques using the proposed method. After that, average result is calculated from the all balance datasets as final result. The suggested approach is depicted in Fig. 1 as an overall technique.

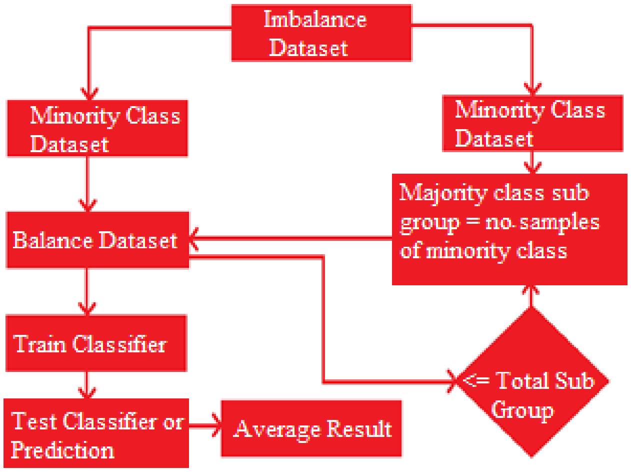


Fig. 1. Flowchart of Sub Group based Blenching Method.

Let's consider an example by considering a dataset of 100 samples where the total number of minority class samples is 20 and total number of majority class samples is 80.

$totalSamples_{minClas} = 20\ and$

$totalSamples_{majClass} = 80;$

The minority class samples forms four balanced dataset by randomly selecting twenty or equal number of samples from the majority class.

$subSampel_{majClass} = 20;$

Now, these minority class dataset and majority class sub datasets are appended to form a complete balanced dataset.

$balanceDataset =$
$append(totalSamples_{minClas}, subSampel_{majClass})$;

Now, a complete balanced dataset of forty samples is formed whereas twenty samples of minority class and the rest twenty samples of majority class. This formation of balanced dataset iterates four times according to the total number of majority class samples is divided by the total number of minority class samples or $\frac{4}{20} = 20$; Finally, the result is calculated and summed for each balanced dataset and average result is considered as the outcome of the proposed method..

Algorithm 1: SGBBM

Data: Imbalance Dataset: DS=1, 2, 3....N // each sample i contains a number of features and corresponding the majority and minority class.

Result: Balance Dataset

```
Procedure SGBBM(DS, N);
//separate the dataset into the majority and minority class dataset.
$dataset_{majClass}$= allSamplesOfTtheMajorityClass;
$dataset_{\text{minClass}}$= allSamplesOfTheMinorityClass;
$totalSamples_{minClass}$= totalNumberOfMinorityClassSamples;
$totalSamples_{majClass}$= totalNumberOfMajorityClassSamples;
$balanceDataset_{sub}$ =’’;
result = 0;
For count ∈ [1, $totalSamples_{minClass}$] do
    a. $subDataset_{majClass} = RandSelect(dataset_{majClass}, totalSamples_{minClass})$;
    //equal of $totalSamples_{majClass}$ samples is selected randomly.
    b. $balanceDataset_{sub}$ = Append($dataset_{\text{minClass}}$ , $subDataset_{majClass}$,);
    // Balance dataset are created
    c. Result+=Prediction ($balanceDataset_{sub}$);
End for
averageResult = $\frac{Result}{dataset_{\text{minClass}}}$;
return (averageResult);
End Procedure
```

IV. Result and Discussion

A. Implementation Methods

1) Random forest: Random Forest is a set of tree predictors that is a supervised learning algorithm that can be used for classification as well as regression, generating a number of classifiers and aggregating their results to achieve the best results. In the random forest, every tree depends on the values of a random vector sampled separately and distributed equally to all trees in the forest [31]. This can handle high dimensional data by building decision trees on randomly selected data samples which are predicted for certain data from each tree. Finally, the best solution is selected by means of voting. It works as follows:

- It chooses random instances from the dataset provided.
- It constructs decision tree for each instance and obtains predictive results from each decision tree.
- It applies the voting system to all predicted results.
- Finally, it chooses the best predicted outcome.

Few important characteristics of RF are as follows [14]:

- It can effectively measure the missing data in the dataset.
- Using weighted rand forest (WRF) process, the error in imbalanced dataset can be balanced.
- The value of variables used in the classification can be calculated.

The Random Forest classifier's full operation flow chart is shown in Fig. 2.

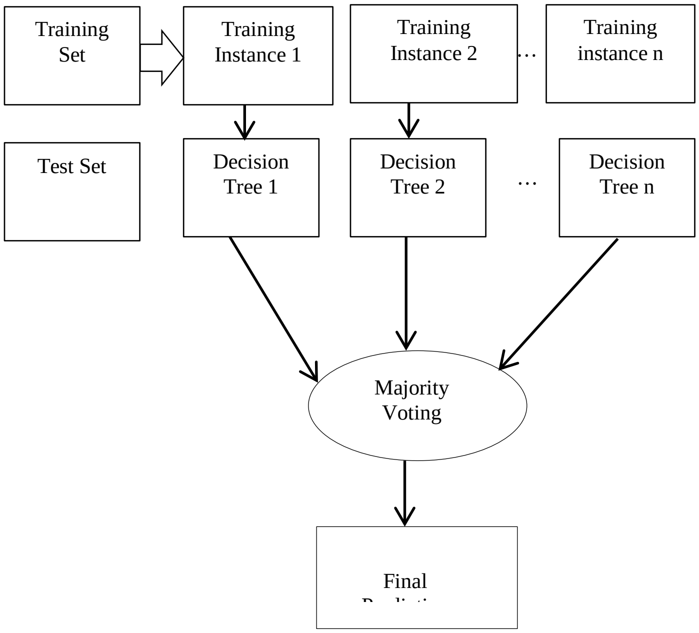


Fig. 2. Random Forest Working Principle.

2) Naïve Bayes: In machine learning, naïve Bayes methods are a series of supervised learning algorithms based on the application of Bayes' theorem, a probabilistic model of machine learning. This is a probabilistic model in which each pair of features is independent of each other provided the value of the class variable to be categorized. It works by translating the dataset into a frequency table and requires a number of linear variables for a linear problem. It comes in the form below:

$$P\left(\frac{A}{B}\right) = \frac{P\left(\frac{B}{A}\right)P(A)}{P(B)} \tag{1}$$

Where, the probability of the A event is determined, while the B occurred. Here, the A is hypothesis and B is evidence. It calculates the probability of each input class and helps predict the target class of the unknown data samples. The general theorem of Bayes uses the following formula to measure the posterior probability for each class.

$$P\left(\frac{C}{X}\right) = P\left(\frac{X}{C}\right)P(C) \quad (2)$$

$$P\left(\frac{C}{X}\right) = P\left(\frac{X_1}{C}\right) \times P\left(\frac{X_2}{C}\right) \times \ldots . \times P\left(\frac{X_n}{C}\right) \times P(C) \quad (3)$$

- $P\left(\frac{C}{X}\right)$ = The corresponding probability of target class in which the predictor attribute is assigned.
- $P(C)$ = The target class's prior probabilities.
- $P\left(\frac{X}{C}\right)$= The probability of the predictor variable in which the target class is given.
- $P(X)$ = The predictor variable's prior probability.

3) K-Nearest neighbor: The K-nearest neighbor method is a non-parametric simple, easy to implement, supervised machine learning technique that classifies the new samples on the basis of similarity measures that can be used for predictive problems of classification and regression. In KNN, three approaches to distance measurements are true only for variables in KNN such as Euclidean, Manhattan, Murkowski. It uses the 'function similarity' to forecast new data point values. If K=1(where k is an integer), the row is then simply allocated to the class of its nearest data point. The KNN does not have a special training process and the entire dataset is used during the classification. Fig. 3 depicts the activity of the KNN.

The KNN algorithm works as follows:

1. Firstly it is needed to take a value of K i.e. the closest data points where K can be any integer.
2. Within the test data set the following steps are performed for each data point:
   a. Measure the distance between the training data and test data in-row using any distance measurement technique, such as Euclidean or Manhattan or Hamming distance, where Euclidean technique is most used.
   b. The rows are ordered in ascending order according to the distance calculated.
   c. Top K rows are picked from the sorted array.
   d. Now the test point is allocated a class according to the most frequent class in this test point row.
3. End.

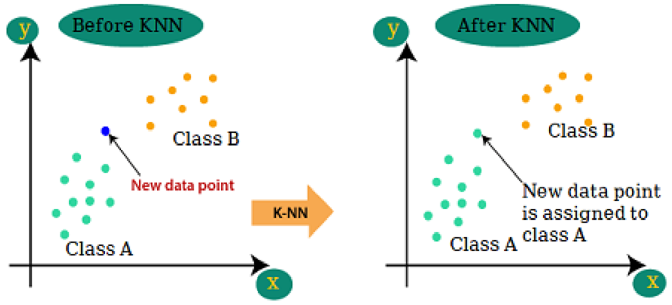


Fig. 3. K-nearest Neighbor Algorithm.

B. Evaluation and Experimental Result

1) Dataset description: The author has been tested this algorithm using three benchmark datasets such as credit card, abalone and wine quality datasets that are presented in the Table I with imbalance ratio (minority: majority) of 1:577, 1:16 and 1:2. The abalone and wine quality datasets were taken from the UCI repository and fetched with imblearch phyton 3.5 library.

The credit card dataset has been collected during research collaboration from ULB's Worldline and Machine Learning Group (ULB University Libre de Bruxelles) on big data processing and fraud detection. The dataset contains 284,807 transactions made by the holders of credit cards in Europe in September 2013.

All the features of the credit card dataset are shown in the Table II are not identical in terms of the distribution the transaction amount and transaction time. In order to build a machine learning based credit card fraud detection model for imbalance dataset, firstly we have prepared raw dataset with the feature values that are mentioned in the Table II. All transactions took place within two days with 492 fraud transactions out of 284,807 transactions where the proportion of the positive class (fraud) of all transactions was 0.172 percent. The dataset includes only the numerical variables such as V_1 to V_n (n=1, 2...28) which are the fundamental components of this dataset.

TABLE I. DATASET DESCRIPTION

	Dataset Name	Description	Minority: Majority	# Samples	# Features
1	Credit Card Dataset	Credit Card fraud detection	492:284315 => (1:577)	284,807	31
2	Abalone Dataset	Prediction of the abalone age	42:689 => (1:16)	731	9
3	Wine Quality Dataset	Prediction the quality of the white wine	175:4898 => (1:27)	5073	13

TABLE II. CREDIT CARD DATASET FEATURES WITH VALUE TYPE

Feature Name	**Value Type**	**Feature Name**	**Value Type**	**Feature Name**	**Value Type**
Time	**Float**	V11	**Float**	V22	**Float**
V1	**Float**	V12	**Float**	V23	**Float**
V2	**Float**	V13	**Float**	V24	**Float**
V3	**Float**	V14	**Float**	V25	**Float**
V4	**Float**	V15	**Float**	V26	**Float**
V5	**Float**	V16	**Float**	V27	**Float**
V6	**Float**	V17	**Float**	V28	**Float**
V7	**Float**	V18	**Float**	Amount	**Float**
V8	**Float**	V19	**Float**	Class	**Integer**
V9	**Float**	V20	**Float**	V22	**Float**
V10	**Float**	V21	**Float**	V23	**Float**

Table III describes the all features of the abalone dataset whereas two features are integer types and rest of the features is float types. The abalone dataset is used for foreseeing the period of abalone from actual estimations. Cutting the shell through the cone, staining it, and counting the number of rings through a microscope are used to calculate the age of abalone.

Table IV lists the characteristics of the wine quality dataset, two of which are integer types and the others are float types.

The red varieties of the Portuguese "Vinho Verde" wine are the subject of this dataset.

2) Experiment setup: In order to evaluate the effectiveness of our proposed method, we aim to answer the following two questions:

- Question 1: Is the proposed machine learning based prediction method able to detect the credit card fraud and to provide significant effectiveness of result for various test cases?
- Question 2: How effective and efficient is our proposed method compared to the existing machine learning based balancing methods?

In answering the above questions, we have conducted experiments on a credit card dataset consisting of two binary classes discussed in a previous section. We have implemented and tested all the methods in Python programming language, in which we have used Scikit-learn, the most popular machine learning library and executed on a Windows PC for predictive data analysis. In the following subsections, we first define the evaluation metrics that are taken into account to evaluate our proposed prediction method and then discuss the results of the experiment which address the above questions defined for this experimental study.

TABLE III. ABALONE DATASET FEATURES WITH VALUE TYPE

Feature Name	Value Type	Feature Name	Value Type	Feature Name	Value Type
Type	**Integer**	Length	**Float**	Diameter	**Float**
Height	**Float**	Whole weight	**Float**	Shucked weight	**Float**
Viscera weight	**Float**	Shell weight	**Float**	Rings	**Integer**

TABLE IV. WINE QUALITY DATASET FEATURES WITH VALUE TYPE

Feature Name	Value Type	Feature Name	Value Type	Feature Name	Value Type
Sex	**Integer**	Fixed Acidity	**Float**	Volatile Acidity	**Float**
Citric Acid	**Float**	Residual Sugar	**Float**	Chlorides	**Float**
Free Sulfur Dioxide	**Float**	Total Sulfur Dioxide	**Float**	Density	**Float**
pH	**Float**	Sulphates	**Float**	Alcohol	**Float**
Quality	**Integer**				

3) Evaluation matric: The evaluation criteria are an important factor in assessing the classification efficiency. In order to measure the effectiveness and efficiency, we take into account the *accuracy*, *specificity*, *precision*, *recall*, *f-score* to test our proposed efficient prediction methodology that are defined as follows

a) Accuracy: The accuracy rate is normally the most common empirical measure in the classification algorithms for machine learning. Accuracy is the ratio of number of accurate predictions to total input samples. Rate of classification or accuracy is determined by the relation:

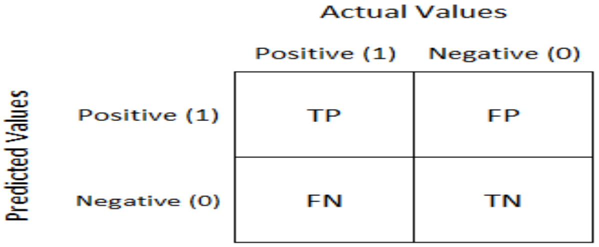
$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{4}$$

In the imbalance data set domain the accuracy rate is not a valid output assessment metric. Since it does not give any result from correctly or incorrectly classified samples of the various classes. This can trigger an incorrect conclusion for this reason. When a classifier achieve a 91 percent accuracy rate is not ideal because it classifies all samples as negative. So the *Confusion metric* is another evaluation metric in the domain of imbalance dataset.

b) Confusion Matrix: A confusion matrix is a description of the predictive results on a classification problem for machine learning. It records the samples for each class correctly and incorrectly predicted. Pizzi et al. [28] discuss the confusion matrix in more detail. The confusion matrix demonstrates how the classification model becomes confused when it makes data set predictions where performance can be two or more classes. It is a table with four different expected and actual combinations of values. This is represented by four pieces of data:

- True Positive (TP): An element is expected to be defective and it is defective. Ultimately it applies to the number of successful instances listed correctly.
- False Positive (FP): An element is expected to be defective and is not defective. This applies to how many derogatory classes are misclassified.
- True Negative (TN): An element is expected not to be defective, and is not defective. This refers to the number of correctly identified negative instances.
- False Negative (FN): An element is expected not to be faulty, and is faulty. This applies to the number of positive instances that are misclassified.

The structure of the confusion matrix is shown in Fig. 4.



	Positive (1)	Negative (0)
Positive (1)	TP	FP
Negative (0)	FN	TN

Fig. 4. Structure of the Confusion Matrix.

The specificity, recall, f-score and precision are defined as follows [32]:

Specificity: $\frac{TN}{TN+FP}$ (5)

Recall: $\frac{TP}{TP+FN}$ (6)

F-score: $\frac{2*TP}{2*TP+FP+FN}$ (7)

Precision: $\frac{TP}{TP+FP}$ (8)

Where TP denotes true positives, TN denotes true negatives, FP denotes false positives and FN denotes false negatives in these above formal equations of specificity, recall, f-score and precision.

4) Experimental result: In order to answer the first question mentioned above, in this experiment show the experimental results of our machine learning based prediction detection method. We have used the three benchmark datasets to verify the performance of the experimental results on different type of data. To calculate the experimental results for various test cases, we first built the method using a subset of 80% data samples from the given highly imbalanced dataset and used the remaining 20% of data samples for testing the proposed method. The experimental results are calculated by generating a confusion matrix that presents the number of true positives, true negatives, false positives and false negatives. According to these values, Table V and Table VI present the prediction results and true & false positive rate of our *proposed* method respectively in terms of specificity, recall, f-score, precision and accuracy for each individual class using the given highly imbalanced dataset in order to show the experimental results. If we observe Table V, we see that for each class, our proposed method gives the significant improved results of the specificity, recall, f-score, precision and accuracy. From Table V, we see that the accuracy, precision, specificity, recall and f-score of Random Forest, Naïve Bayes, K-Nearest Neighbor are (99%,96%,90%),(98%, 98%,97%),(86%,93%,85%),(90%,70%,97%) and (92%, 70%, 97%), respectively. If we observe the Table VI, the true positive rate of the Random Forest, Naïve Bayes and K-Nearest Neighbor classifiers using our proposed method are 86%, 93%, 85%, respectively that are very close to the maximum value 1 and the false positive rate are 5% , 72%, 5% respectively that are very lower than the existing methods. In this way, from overall experimental results shown in Table V and Table VI, we can say that our proposed machine learning based prediction method in terms of highly imbalanced dataset is able to efficiently detect either fraud or not fraud class according to their occurring patterns in the highly imbalanced credit card fraud dataset and consequently provides a significant effectiveness for a various test cases.

Table VII and Table VIII represent the observe values for Specificity, Recall, F-score, Precision, Accuracy, True Positive rate and False Positive rate using the abalone dataset. Random Forest, Nave Bayes, and K-Nearest Neighbor have almost as high a precision as the most advanced algorithms. The Nave Bayes and KNN recalls are very similar to one, though the Random Forest recall is slightly lower. The F-score of the KNN is very close to 1, while the F-scores of Random Forest and Nave Bayes are a little lower but still appropriate.

For all base line algorithms, the True Positive Rate (TPR) is significantly higher than the False Positive Rate (FPR), which is significantly lower.

The experimental results of the proposed method on a wine quality dataset using various machine learning algorithms are shown in Tables IX and X. Random Forest and Nave Bayes have substantially high accuracy to the highest accuracy, while KNN has a satisfactory and better accuracy of 91%.

The Random Forest algorithm has a very high specificity, but it performs better than other Nave Bayes and KNN algorithms.

The True Positive Rate (TPR) for Random Forest, Nave Bayes, and KNN is significantly higher, while the False Positive Rate (FPR) is significantly lower, even though the FPR of Nave Bayes is 91 percent, suggesting that the Nave Bayes' performance in terms of FPR is not good.

TABLE V. EFFECTIVENESS COMPARISON OF THE DIFFERENT CLASSIFIERS USING PROPOSED ALGORITHM ON CREDIT CARD DATASET

#	Classifier	Specificity	Recall	F-score	Precision	Accuracy
1	Random Forest	98%	86%	90%	92%	99%
2	Naïve Bayes	98%	93%	70%	70%	96%
3	K-Nearest Neighbor	97%	85%	90%	97%	90%

TABLE VI. COMPARISON OF TRUE POSITIVE RATE VERSUS FALSE POSITIVE RATE OF DIFFERENT CLASSIFIERS USING PROPOSED ALGORITHM ON CREDIT CARD DATASET

#	Classifier	True Positive Rate (TPR)	False Positive Rate (FPR)
1	Random Forest	86%	5%
2	Naïve Bayes	93%	72%
3	K-Nearest Neighbor	85%	5%

TABLE VII. EFFECTIVENESS COMPARISON OF THE DIFFERENT CLASSIFIERS USING PROPOSED ALGORITHM ON ABALONE DATASET

#	Classifier	Specificity	Recall	F-score	Precision	Accuracy
1	Random Forest	97%	94%	95%	95%	97%
2	Naïve Bayes	96.2%	98%	94%	92%	98.23%
3	K-Nearest Neighbor	98%	98%	98%	97%	94%

TABLE VIII. COMPARISON OF TRUE POSITIVE RATE VERSUS FALSE POSITIVE RATE OF DIFFERENT CLASSIFIERS USING PROPOSED ALGORITHM ON ABALONE DATASET

#	Classifier	True Positive Rate (TPR)	False Positive Rate (FPR)
1	Random Forest	89%	10%
2	Naïve Bayes	96%	81.4%
3	K-Nearest Neighbor	98.2%	15.6%

TABLE IX. EFFECTIVENESS COMPARISON OF THE DIFFERENT CLASSIFIERS USING PROPOSED ALGORITHM ON WINE DATASET

#	Classifier	Specificity	Recall	F-score	Precision	Accuracy
1	Random Forest	98.6%	91%	95.3%	93.4%	99%
2	Naïve Bayes	93%	83%	90%	89%	98%
3	K-Nearest Neighbor	91%	92%	92.45%	95%	91%

TABLE X. COMPARISON OF TRUE POSITIVE RATE VERSUS FALSE POSITIVE RATE OF DIFFERENT CLASSIFIERS USING PROPOSED ALGORITHM ON WINE DATASET

#	Classifier	True Positive Rate (TPR)	False Positive Rate (FPR)
1	Random Forest	94%	12%
2	Naïve Bayes	97.6%	91%
3	K-Nearest Neighbor	96.5%	10%

5) Effectiveness comparison: In order to answer the second question, in this experiment, we calculate and compare the effectiveness of our proposed method with the existing algorithm i.e. *SMOTE*. To show the effectiveness of different machine learning based models, we first select several popular baseline algorithms such as Random Forests (RF), Naïve Bayes (NB) and K-Nearest Neighbor (KNN) for the sake of effectiveness comparisons. For each algorithm, we calculate the experimental results using the same highly imbalanced dataset, in order to compare the model fairly. To compute the effectiveness of different baseline algorithms, we see that Table V and Table XI show the relative comparison of the experimental results of different models using t our proposed method and SMOTE respectively in terms of accuracy, precision, specificity, recall and f-score on credit card dataset. For each baseline model, we use the same training and testing sets of data, where 80% of data are used to train the model and the rest 20% data are used for testing the model. Our proposed model's specificity for all machine learning algorithms used here are significantly higher than the specificity of the SMOTE, indicating superior performance on the credit card dataset. The recalls of proposed method are also better than the SMOTE. The proposed method's f-scores are considerably higher, while the f-score of Nave Bayes has plummeted. The proposed method outperforms the traditional SMOTE method in not only specificity, recall, and f-score, but also in all output matrixes.

If we consider Table VII and Table XII, the effectiveness of different baseline models using our proposed methods is better than the effectiveness of different baseline models using *SMOTE* method on abalone dataset. Using the proposed approach on the abalone dataset, the accuracy for all machine learning algorithms used here is substantially higher than the SMOTE. On the abalone dataset, the recall of Nave Bayes and KNN using the proposed model is nearly 100 percent higher than that of SMOTE, demonstrating the proposed method's superior efficiency whereas the f-score, precision and accuracy are still better than SMOTE. Because all samples of the imbalanced dataset are used in the experiment. As a result, the data redundancy and removal of important sample from the dataset are solved successfully.

Table IX and Table X show the significant differences from the result of the proposed method to SMOTE method on wine quality dataset that proofs the robustness of the proposed method. The proposed method improves the accuracy of the Random Forest and Nave Bayes to a maximum of 100% compared to the standard SMTOE on wine quality dataset. In this regard, the proposed approach not only increases the Random Forest's specificity to the nearest 100 percent, but also greatly improves the recall, f-score, precision, and accuracy output values.

On all three datasets, the proposed method outperforms than the conventional SMOTE method in terms of specificity, recall, f-score, precision, and accuracy, as seen in the above effectiveness comparison. The proposed SGBB has greater generalization capabilities than SMOTE, as shown by the better performance of all evaluation matrices. Since the proposed method eliminates all of the above-mentioned shortcomings of SMOTE, it can be used in all complex cases to predict classes due to its superior performance over conventional SMOTE.

Fig. 5 and 6 show the comparative results of different classifiers that are obtained after experimenting by authors using SMOTE and proposed algorithm respectively. In terms of specificity, recall and accuracy our proposed algorithm is much better than the SMOTE. The F-score of the Naïve Bayes and KNN are drastically down in SMOTE whereas the F-score of these classifiers are efficiently getting higher in our proposed algorithm. The precision of Naïve Bayes and KNN are 1% and 34% using SMOTE whereas these values are 70% and 97% respectively in our proposed algorithm that is very much high.

Fig. 7 and 8 show the true and false positive rates of the SMOTE and our proposed algorithm. The TPR of Random Forest using our proposed algorithm is getting higher than the SMOTE whereas the TNR is 99% that is maximum and unexpected in machine learning. On the other hand, the TPR and TNR of Naïve Bayes are being greater and lower respectively using our proposed algorithm than the SMOTE whereas the TPR of Naïve Bayes is being fallen drastically down to 2% using SMOTE. The TPR and TNR of KNN are greater and lower than the SMOTE respectively that is better performance of the algorithm. The TNR is maximum to 99% using SMOTE and the TPR is much lower than the proposed algorithm that indicates the less performance in machine learning.

For the wine dataset in Fig. 9 and 10, the positive predictive value is considerably greater in terms of Random Forest algorithm using the proposed SGBBA. On the other hand, positive predictive value falls down then the negative predictive value using the SMOTE. In terms of Naïve Bayes, the positive predictive value is still higher than the negative predictive value using the SGBBA whereas the SMOTE drastically falls down at 22.4% although the negative predictive value is too much higher that is another pitfall of the SMOTE. The proposed SGBBA consistently performs well during the prediction of positive value using the KNN whereas the SMOTE performs lower.

In the abalone dataset, the positive predictive value of Random Forest is efficiently higher whereas the negative predictive value is 10% using the SGBBA. On the other hand, the negative predictive value is higher than the positive predictive value using SMOTE. Similarly, the SGBBA performs very well in terms of positive predictive value in Naïve Bayes and KNN classifiers than the SMOTE that are showing in Fig. 11 and Fig. 12.

The proposed method's superior performance in all evaluation matrices demonstrates the robustness of the imbalance dataset handling method, which can be applied to any imbalance dataset for making balance dataset in machine learning. Thus, our proposed method not only remove the data redundancy, removal of important data samples and but also increase the prediction results for various test cases. Therefore, according to the experimental results shown in Table V, Table VI, Table VII, Table VIII, Table IX, Table X, Table XI, Table XII, Table XIII, Fig. 5, and Fig. 7 and above experimental result analysis, we can conclude that our proposed method is more effective and efficient than the existing SMOTE method during the experiment of data analysis and machine learning using the highly imbalanced dataset.

TABLE XI. EFFECTIVENESS COMPARISON OF THE DIFFERENT CLASSIFIERS USING SMOTE ALGORITHM ON CREDIT CARD DATASET

#	Classifier	Specificity	Recall	F-score	Precision	Accuracy
1	Random Forest	91%	83%	84%	85%	91%
2	Naïve Bayes	95%	90%	1%	1%	90%
3	K-Nearest Neighbor	90%	82%	49%	34%	85%

TABLE XII. EFFECTIVENESS COMPARISON OF THE DIFFERENT CLASSIFIERS USING SMOTE ALGORITHM ON ABALONE DATASET

#	Classifier	Specificity	Recall	F-score	Precision	Accuracy
1	Random Forest	95.32%	88.35%	92%	88%	93%
2	Naïve Bayes	91.12%	93.2%	84%	76%	91.5%
3	K-Nearest Neighbor	94.3%	94.3%	92%	87%	90%

TABLE XIII. EFFECTIVENESS COMPARISON OF THE DIFFERENT CLASSIFIERS USING SMOTE ALGORITHM ON WINE QUALITY DATASET

#	Classifier	Specificity	Recall	F-score	Precision	Accuracy
1	Random Forest	93%	83%	90%	89%	98%
2	Naïve Bayes	92%	90%	95.5%	82%	94%
3	K-Nearest Neighbor	91%	92%	49%	95%	91%

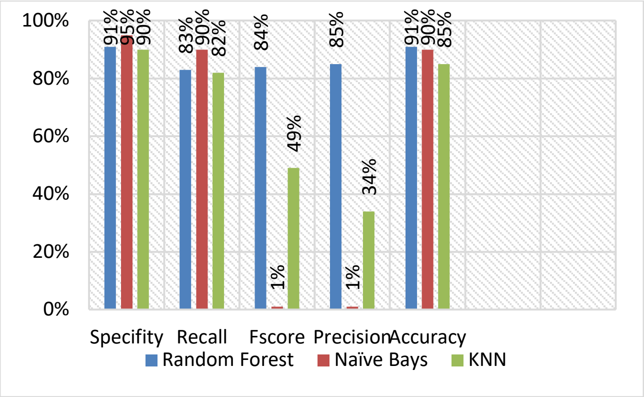


Fig. 5. Accuracy Comparison of the different Classifiers using SMOTE on Credit Card Dataset.

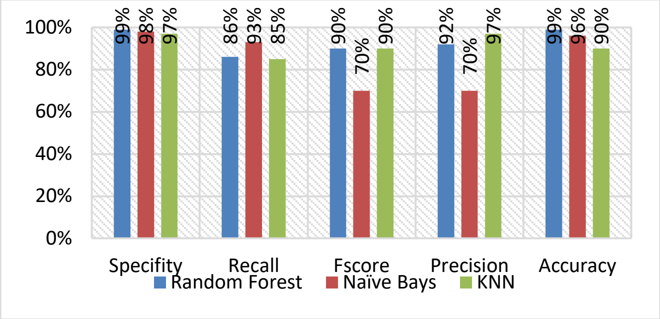


Fig. 6. Accuracy Comparison with Proposed Approach on Credit Card Dataset.

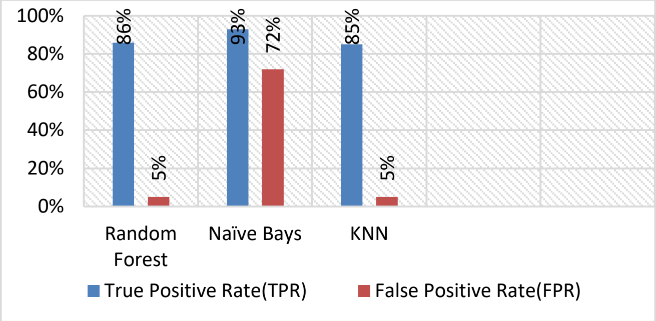


Fig. 7. Comparison of True Positive Rate versus True Negative Rate of different Classifiers using Proposed Algorithm on Credit Card Dataset.

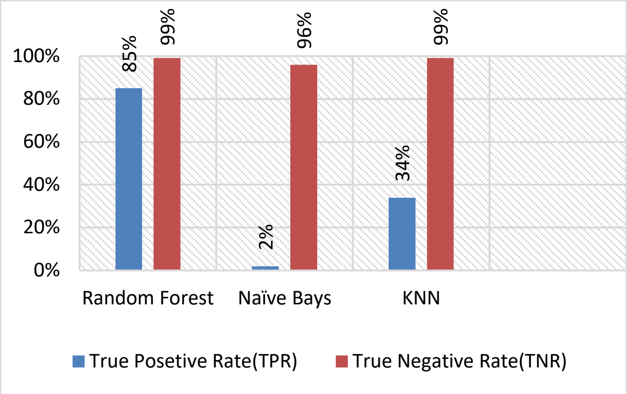


Fig. 8. True Positive Rate versus True Negative Rate of different Classifiers using SMOTE Algorithm on Credit Card Dataset.

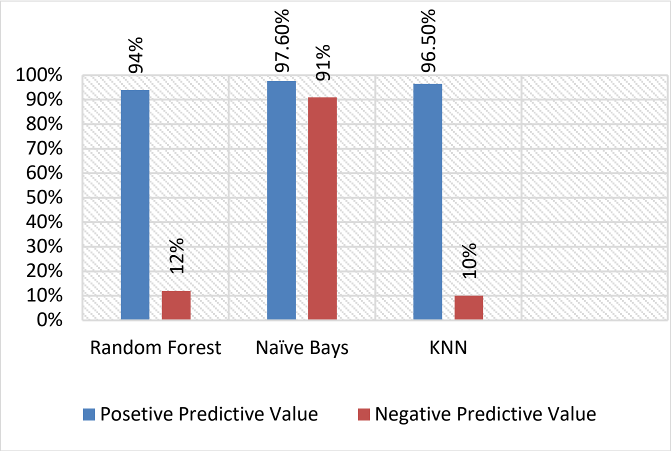


Fig. 9. True Positive Rate versus True Negative Rate of different Classifiers using Proposed Algorithm on Wine Dataset.

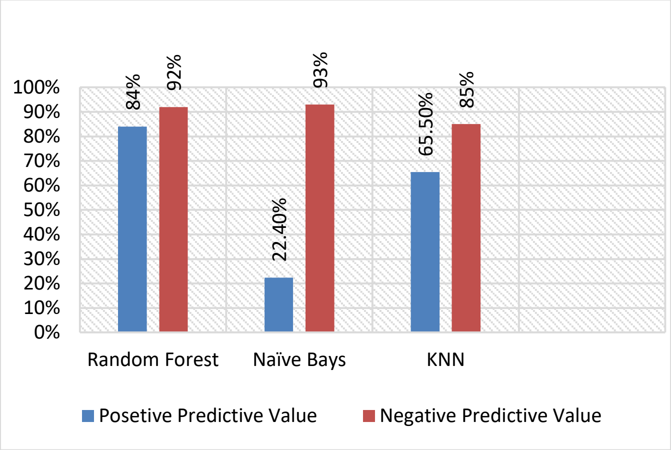


Fig. 10. True Positive Rate versus True Negative Rate of different Classifiers using SMOTE Algorithm on Wine Dataset.

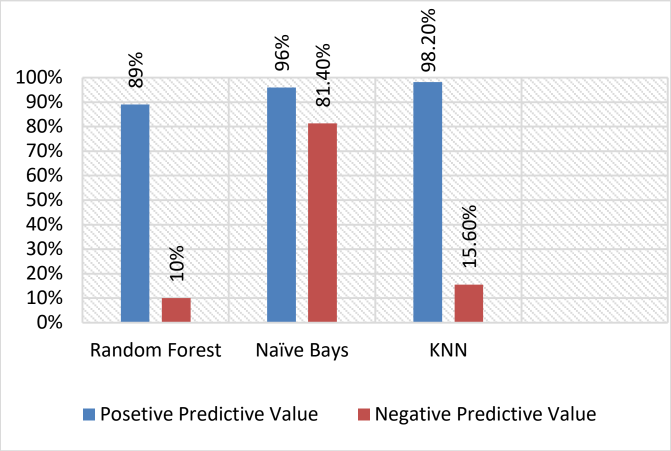


Fig. 11. True Positive Rate versus True Negative Rate of different Classifiers using Proposed Algorithm on Abalone Dataset.

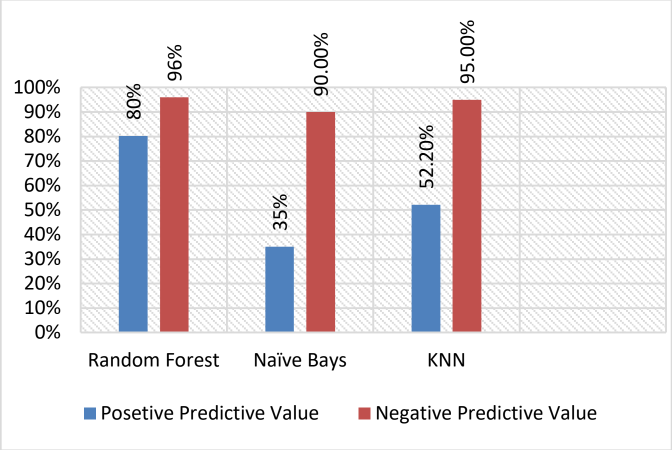


Fig. 12. True Positive Rate versus True Negative Rate of different Classifiers using SMOTE Algorithm on Abalone Dataset.

V. Conclusion and Future Work

Trying to the train of model using imbalance dataset by researcher is a challenging task due to biased of class in the dataset. The biasing of the classes in the dataset decreases the performance of the classifiers in an amount. As a result perfect prediction is not possible in the imbalance dataset. The processing of data from an imbalanced dataset is a key challenge and activity in data science and machine learning. Because without having a balanced dataset the creation of a new classification model produces a skewed prediction result to the majority class ignoring the minority class. The prediction from the imbalanced dataset is therefore unnecessary and useless for machine learning. A variety of methods are explored in the background analysis section for creating a balanced dataset from the imbalance dataset. Each methodology has a serious problem with balancing data sets such as data redundancy and removal of essential samples of data. To solve this problem we have introduced an algorithm named – SGBBA: An efficient algorithm for prediction system in machine learning using Imbalance dataset where each sub dataset is a balanced dataset without any data redundancy and removal of important data sample. This new algorithm is implemented with three different classification algorithms and their results are compared with the predictive result of SMOTE algorithm. The three datasets have played important rule in terms of determining the efficiency and performance of the proposed algorithm. Our approach has the following advantages:

- It solves the redundancy of data samples from existing methods.
- It solves the elimination of significant samples problem from the original dataset.

Such benefits have advantages for researchers, practitioners and reviewers so that they can use this theoretical model to predict results in terms of machine learning imbalance datasets.

Our future goal is to develop an efficient minority class based algorithm for a prediction system in machine learning with optimal features of the dataset.

Abbreviations

N_{min}: Total number of samples of the minority class, N_{max} : Total number of samples of the majority class, SMOTE : Synthetic Minority Over-sampling Technique , TP : True Positive, TN : True Negative, FN: False Negative, FP : False positive.

Acknowledgments

We are extremely grateful to all co-authors for their strong feedbacks and contributions in the field of learning from imbalanced datasets whereas 1. Saiful Islam collected data, studied and implemented the proposed architecture. 2. Umme Sara contributed to designing the idea, writing the proposed methodology partly and compiling the article. 3. Abu Kawsar contributed to the drawing of flow chart and result representation graph. 4. Anichur Rahman reviewed the introduction and organizing the article. 5. Depanjali Kundu. 6. Diganta Das Dipta reviewed background knowledge and wrote the experiment part. 7. A.N.M. Rezaul Karim contributed to the mathematical analysis of the proposed method. 8. Mahedi Hasan contributed in main algorithm for dataset balancing process. We are pleased to work with them and to contribute to this domain.

References

[1] Everlandio R.Q. Fernandes, Andre C.P.L.F. de Carvalho and Xin Yaho. Ensemble of Classifiers based on Multi Objective Genetic Sampling for Imbalanced Data. Journal of LATEX Class Files, VOL.14, No. 8, August 2015.

[2] Guo Haixiang, Li Yijing, Jennifer Shang, Gu Mingyun, Huang Yuanyue, Gong Bing. Learning from class-imbalanced data: Review of methods and applications. Expert Systems With Applications 73 (2017) 220-239.

[3] Wei-Chao Lin, Chinh-Fong Tsai, Ya-Han Hu, Jing-Shang Jhang. Clustering –based undersampling in class-imbalance data. Information Science 409-410 (2017) 17-26.

[4] Mohammed Khalilia, Sounak Chakraborty and Mihal Popescu. Predicting disease risks from highly imbalanced data using random forest. Khalilia et al. BMC Medical Informatics and Decision Making 2011.

[5] Nitesh V. Chawla. Data Mining for Imbalanced Datasets: An Overview. Data Mining Knowledge Discovery Handbook, 2nd ed. DOI10.1007/978-0-387-09823-4_45.

[6] Mohd Farizul Mat Ghani. Intelligent Heart Disease Prediction System Using Data Mining Techniques.IJCSNS International Journal of Computer Science and Network Security, VOL. 8 No. 8, August 2008.

[7] R.E. Schapire , The strength of weak learnabilty, Mach. Learn. 5 (2) (1990) 197–227.

[8] L. Breiman , Bagging predictors, Mach. Learn. 24 (2) (1996) 123–140.

[9] N.V. Chawla , A. Lazarevic , L.O. Hall , K.W. Bowyer , SMOTEBoost: improving prediction of the minority class in boosting, in: European Conference on Principles and Practice of Knowledge Discovery in Databases, 2003, pp. 107–119.

[10] Seiffert , T. Khoshgoftaar , J. Van Hulse , A. Napolitano , RUSBoost: a hybrid approach to alleviating class imbalance, IEEE Trans. Syst. Man Cybern. –Part A 40 (1) (2010) 185–197.

[11] Wang , X. Yao , Diversity analysis on imbalanced data sets by using ensemble models, in: IEEE International Symposium on Computational Intelligence and Data Mining, 2009, pp. 324–331.

[12] Barandela , R.M. Valdovinos , J.S. Sanchez , New applications of ensembles of classifiers, Pattern Anal. Appl. 6 (2003) 245–256.

[13] N.V. Chawla , A. Lazarevic , L.O. Hall , K.W. Bowyer , SMOTE Boost: improving prediction of the minority class in boosting, in: European Conference on Principles and Practice of Knowledge Discovery in Databases, 2003, pp. 107–119.

[14] Breiman L: Random forests. Machine learning. 2001, 45 (1): 5-32. 10.1023/A:1010933404324.

[15] Seiffert , T. Khoshgoftaar , J. Van Hulse , A. Napolitano , RUSBoost: a hybrid approach to alleviating class imbalance, IEEE Trans. Syst. Man Cybern. –Part A 40 (1) (2010) 185–197.

[16] Barandela , R.M. Valdovinos , J.S. Sanchez , New applications of ensembles of classifiers, Pattern Anal. Appl. 6 (2003) 245–256.

[17] Galar , A. Fernandez , E. Barrenechea , H. Bustince , F. Herrera , A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches, IEEE Trans. Syst. Man Cybern. –Part C 42 (4) (2012) 463–484.

[18] Japkowicz, N. (2000a). The Class Imbalance Problem: Significance and Strategies. In Proceedings.of the 2000 International Conference on Artificial Intelligence (IC-AI'2000): Special Track on Inductive Learning, Las Vegas, Nevada.

[19] M. Galar , A. Fernandez , E. Barrenechea , H. Bustince , F. Herrera , A review on ensembles for the class imbalance problem: bagging-, boosting-, and hybrid-based approaches, IEEE Trans. Syst. Man Cybern. –Part C 42 (4) (2012) 463–484.

[20] N.V. Chawla , A. Lazarevic , L.O. Hall , K.W. Bowyer , SMOTEBoost: improving prediction of the minority class in boosting, in: European Conference on Principles and Practice of Knowledge Discovery in Databases, 2003, pp. 107–119.

[21] S. Hu,Y.Liang, L.Ma, and Y.He,"Msmote:Improving classification performance when training data is imbalanced," in Computer Science and Engineering,2009.WCSE `09.Second International Workshop on, vol.2,Oct 2009,pp.13-17.

[22] H.He, Y.Bai, E. Garcia, S.Li et al.,"Adasyn: Adaptive synthetic sampling approach for imbalanced learning " in Neural Networks, 2008.IJCNN 2008. (IEEE World Congress on Computational Intelligence). IEEE International Joint Cnference on. IEEE,2008,pp.1322-1328.

[23] Y.Sun, A. K. C. Wong and M. S. Kamal,"Classification of imbalanced data: a review," IJPRAI, vol. 23, no. 4, pp.687-719, 2009.

[24] M. Kubat and S. Matwin, "Addressing the curse of imbalanced training sets: One-Sided selection," in In Proceedings of the Fourteenth International Conference on Machine Learning. Morgan Kaufmann, 1997,pp.179-186.

[25] B. Schlkopf, J. C Platt, J. Shawe-Taylor, A. J. Smola, and R. C. Williamson, "Estimating the support of a high-dimensional distribution ." Neural Computation, vol. 13, no. 7,pp.1443-1471,2001.

[26] M. Lin, K. Tang, and X. Yao,"Dynamic sampling approach to training neural networks for multiclass imbalance classification." IEEE Trans. Neural Netw. Learning Syste.,vol 24, no. 4, pp.647-660, 2013.

[27] G.E.A .P.A . Batista , R.C. Prati , M.C. Monard , A survey of the behavior of several methods for balancing machine learning training data, SIGKDD Explor. 6 (1) (2004) 20–29.

[28] Y. Sun , A.K.C. Wong , M.S. Kamel , Classification of imbalanced data: a review, Int. J. Pattern Recognit. Artif. Intell. 23 (4) (2009) 687–719.

[29] N. Pizzi, A. Summers, and W. Pedrycz. Software quality prediction using median-adjusted class labels. In Neural Networks, 2002. IJCNN '02. Proceedings of the 2002 International Joint Conference on, volume 3,pages 2405 {2409, 2002.

[30] Charles X. Ling and Chenghui Li.Data Mining for Direct Marketing: Problems and. Solutions.

[31] Leo Breiman, Random Forests. Machine Learning, 45, 5–32, 2001.

[32] Han, J.; Pei, J.; Kamber, M. Data mining: Concepts and Techniques; Elsevier: Amsterdam, The Netherlands, 2011.