# Effective Resampling Approach for Skewed Distribution on Imbalanced Data Set

Mar Mar Nwe and Khin Thidar Lynn

***Abstract*—Accurate classification of unknown input data for imbalanced data sets is difficult, because the predictions of learning classifiers tend to be biased towards the majority class and ignore the minority class. Moreover, the class distribution of imbalanced data has a significant impact on the misclassification rate of the learning classifier. So, this paper introduces an effective data pre-processing approach to improve the efficiency of imbalanced data classification, focusing on the skewed distribution of data points in the imbalanced data set. This proposed approach involves over-sampling and under-sampling techniques based on k-means clustering to overcome the problems associated with imbalanced learning of small disjuncts and small sample size. And, Tomek Link-based under-sampling method is also incorporated into the proposed cluster-based resampling methods to solve the class overlapping problem by eliminating the majority samples in overlapping regions. Experiments are performed on the 25 standard imbalanced data sets by applying four learning classifiers, and validated with the three popular metrics (i.e. Area Under the Curve (AUC), Geometric Mean (G-mean) and Balanced Accuracy (BA)). Specially, we show that the proposed approach has outperformed the other state-of-the-art resampling methods using performance metrics, probabilistic estimation, statistical analysis and multi-criteria decision-making methodology (MCDM).**

***Index Terms*—Imbalanced Data Classification, Under-Sampling, Over-Sampling, Ensemble Learning, Probabilistic Estimation, Multi-Criteria Decision-Making.**

## I. Introduction

IN the fields of data mining and machine learning, most learning algorithms are designed to improve the overall classification accuracy, and assumed that the number of samples in different classes is equal. As a result, these algorithms do not effectively handle the class imbalanced data sets [15]. The class imbalanced data set refers to a data set where the size of a class is larger than the size of another class (i.e. majority class and minority class, respectively) [10], [17]. Without considering the problem of class imbalance, the performance of learning algorithms is dominated by the majority class samples and the minority class samples are ignored as noise.

Nowadays, the class-imbalanced data sets are available in various real-world applications such as prediction of natural disaster, prediction of software defect, prediction of bankruptcy, prediction of rare disease in medical diagnosis and fraudulent credit card transaction [9], [10], [15], [19], [30], [51], where the misclassification costs for minority class are more interesting and important than the majority class [1], [13], [33]. E.g., In cancer diagnosis, cancer patients and non-cancer patients can be categorized as the minority class and majority class, which means that some patients rarely have cancer disease and most patients suffer from common diseases. So, the prediction of cancer patients can be difficult to predict accurately. Besides, the incorrect prediction on a medical diagnosis not only endangers the patients, but also requires additional resources for the right treatments.

So, to overcome the class imbalance problems, many heuristic approaches have been proposed based on the solutions of data level and algorithm level [11], [15], [46].

***(1) Data Level Solution***: Data level solution is a resampling technique that modifies the class distribution of the training set into a balanced class distribution before applying the learning algorithms [35]. In imbalanced learning, changing the class distribution is more efficient than modifying the learning classifiers, as various classifications can be easily combined [30], [32], [50].

***(2) Algorithmic Level Solution***: The algorithmic level solution involves the creation of new learning models or modification of existing ones without considering class distributions of the training set (e.g. assigning weights to samples for training). This technique requires special knowledge from the relevant classifiers and application domains, to find out why classification is not achieved when the class distribution is imbalanced [17], [27]. Therefore, the main focus of data level and algorithmic level solutions is to improve the efficiency of learning algorithms, affecting the minority class.

Although several solutions related to the class imbalanced learning have been proposed, most of them focused on the class distributions of the training data set. Moreover, the skewed distribution in the imbalanced data sets will lead to the degradation of predicting results. So, the skewed distribution of data points is a special challenge for the learning classifiers. Small disjuncts, class overlapping, and small sample size are the characteristics of skewed distribution that are difficult to achieve the precise results for future predictions [15], [17], [30], [32].

***(1) Small Disjuncts***: The problem of small disjuncts occurs when the minority samples are scattered as several clusters and some clusters are surrounded by the majority class samples. So, the small disjuncts problem is closely related to the within-class imbalance problem, which indicates a significant reduction in the performance of learning classifiers [24], [32], [42].

***(2) Class Overlapping***: The problem of class overlapping occurs when the data samples of different classes have similar data features. If there is an overlap between the samples of different classes, most classifiers wrongly classify the minority class samples into the majority class samples [9], [32], [45].

***(3) Small Sample Size***: The problem of small sample size may be identified if the number of minority class samples is not adequately included in the training data set. This problem

Mar Mar Nwe is with Data Mining and Machine Learning Lab, University of Computer Studies, Mandalay, Myanmar, E-mail: marmarnwe@ucsm.edu.mm

Khin Thidar Lynn is with Faculty of Information Science Department, University of Computer Studies, Mandalay, Myanmar, E-mail: khinthidalynn@ucsm.edu.mm

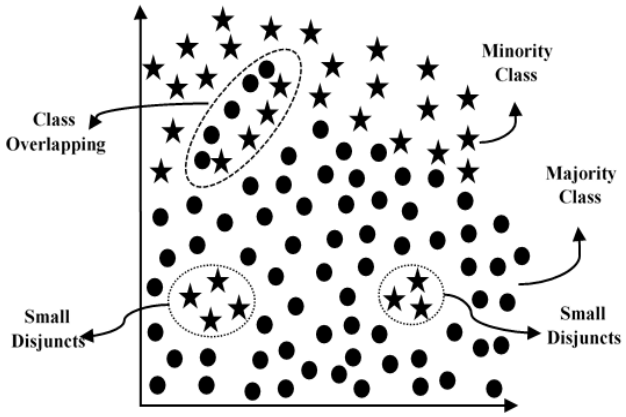


Fig. 1. Class Distributions of Imbalanced Data Set

is closely related to the between-class imbalance problem. The number of samples in different classes is an important part to improve the performance of learning algorithms [15], [19].

Fig. 1 shows the class distribution of an imbalanced data set with majority class samples (denoted by "cycle") and minority class samples (denoted by "star") that makes the classification difficulties. In the next section, the relevant literature review will be discussed, focusing on the characteristics of skewed class distributions and various resampling techniques, as these characteristics are important for improving the efficiency of learning algorithms.

In this paper, we propose an effective data pre-processing approach based on the under-sampling and over-sampling techniques to overcome this skewed distribution problem of an imbalanced data set. In this effective resampling approach, k-means clustering method is combined with Synthetic Minority Over-Sampling Technique (SMOTE) [8] and Random Under-Sampling Technique (RUS) [3] to handle small disjuncts and small sample size problems. Furthermore, Tomek Link-based under-sampling step [43] is integrated into the proposed cluster-based resampling steps to eliminate the majority samples from the overlapping regions. After executing the proposed data pre-processing approach, the new training set is applied by embedding the learning classifiers (including both ensembles and single classifiers) for the model construction. And, the effectiveness of our proposed approach is validated with the performance metrics, probabilistic estimates, and statistically significant tests. Finally, the best sampling method each learning classifier will be determined by implementing TOPSIS (Technique for Preference Order by Similarity to Ideal Solution) approach [38], [40].

The main contributions of this paper are as follows:

1) First, we demonstrate the effectiveness of the skewed distribution in imbalanced data sets by comparing five state-of-the-art resampling techniques using the four learning classifiers.
2) Second, we validate the performance of our proposed approach not only by the evaluation metrics for the class label prediction, but also by the probabilistic estimates for the expected cost, which have never been done before.
3) Finally, TOPSIS approach is implemented to suggest the best sampling method for each learning classifiers.

The rest of the paper is organized as follows. Section 2 describes some popular sampling methods relating to this research. The model construction and the architecture overview of the proposed approach to studying the skewed distribution of imbalanced data set are introduced in Section 3. And, Section 4 displays the proposed approach's experimental settings. Experimental results and analysis of the proposed data pre-processing approach and comparison with other methods are presented in Section 5. Finally, concluding remarks and some suggestions of future works are drawn in the final section.

## II. Related Works

In most studies related to the solution of the class imbalance problem, the use of data level solution has a higher prospective effect because it can improve the performance of the learning classifier for the skewed distribution of imbalanced data [34]. Data level solution is a re-sampling technique that modifies the class distribution of the training set to a balanced class distribution before applying the learning algorithms [30], [35]. This section briefly reviews the related works of imbalanced learning which emphasize the methods of the data level solution. The data level solution can generally be divided into two groups [2]: over-sampling of minority class and under-sampling of majority class.

Over-sampling involves duplicating samples of the minority class or synthesizing new samples of the minority class from existing samples. Random Over-Sampling (ROS) [3] is a popular over-sampling method, which randomly oversamples the minority class samples to balance the class distributions [19]. Although ROS method can produce a balanced class distribution, the training computation's complexity and the over-fitting problem may be affected as disadvantages. To avoid the over-fitting problem, SMOTE method [8] is used to generate the new minority class samples, but it can cause the overlapping problem between different classes [21]. So, to improve the performance of learning classifiers, many changes had been proposed for SMOTE. For example, Borderline-SMOTE over-samples the minority samples near the borderline rather than all samples of a minority class [18]. Safe-level-SMOTE is an improved algorithm of Borderline-SMOTE that identifies the relative area of the new synthetic samples [7].

Besides, the noise and borderline samples are very dangerous for the classification performance [35]. The noise and borderline samples give the class overlapping problem for the learning algorithms. So, SMOTE-NCL, SMOTE-ENN, and SMOTE-TL are used to prevent class overlapping and imbalanced problems with an excessive generalization of SMOTE and dangers of overlapping problems [3], [42]. SMOTE-IPF not only over-samples the borderline samples of the minority class, but also introduces the noise filtering technique to remove the noisy samples in the majority and minority classes [36].

Under-sampling involves deleting samples from the majority class. Random Under-Sampling (RUS) method [3] is a popular under-sampling method involving the random removal of majority class samples to balance the class distribution [19]. Although RUS method can create a balanced class

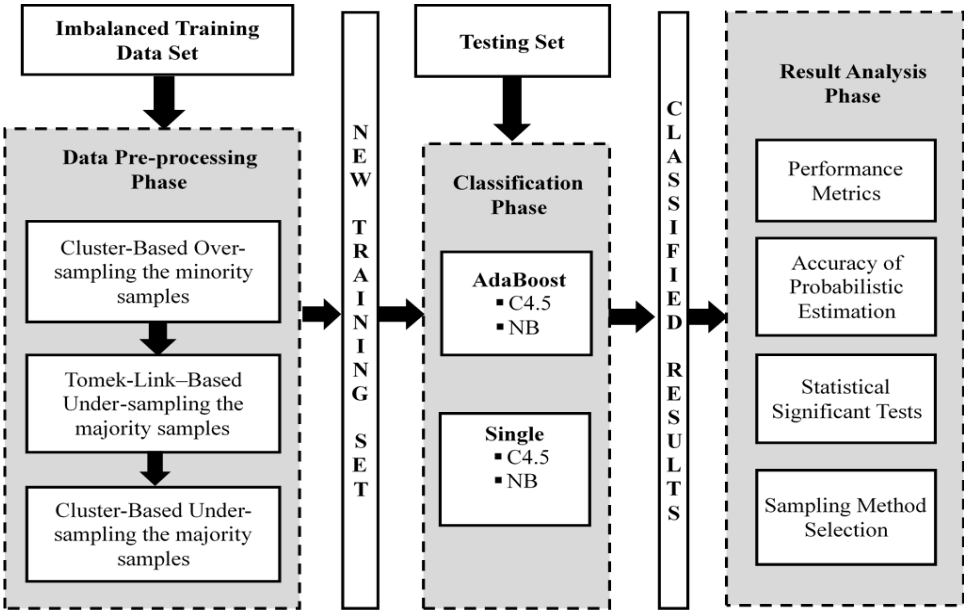


Fig. 2. Framework of Proposed Approach

distribution, it has a useful information loss problem. The cluster-based under-sampling method is also used to achieve uniform the class distributions and to avoid the information loss problem by eliminating the majority samples [30], [41], [50]-[51]. This cluster-based under-sampling method improves the classification performance rather than Random Under-Sampling (RUS) method because it can reduce the useful information loss problems [30].

Furthermore, the classification performance of the imbalanced data set may deteriorate not only the problem of class imbalance, but also the problem of class overlapping. So, the class overlapping problem is also a key factor in the imbalanced data set, because it can be difficult to correctly classify the minority class samples. Earlier studies had been combined the popular data cleaning methods and resampling methods to overcome the class imbalance and overlapping problems [9], [11], [32], [41].

Tomek Link is a popular under-sampling method for solving the class overlapping problem, that has been modified by Condensed Nearest Neighbors (CNN) [22]. In Tomek Link, $x_j$ and $x_i$ are the two samples of different classes (the majority class and minority class, respectively) [11], [41], and d($x_j$ , $x_i$) is the distance between $x_j$ and $x_i$. d($x_j$ , $x_i$) pair is called a Tomek Link if there is no sample $x_g$, such that d($x_j$ , $x_g$) $<$ d($x_j$ , $x_i$) or d($x_i$ , $x_g$) $<$ d($x_j$ , $x_i$). Edited Nearest Neighbor method (ENN) [49] was also used to remove the samples of the majority class by considering its $k$ nearest neighbors that belonging to the other classes, where $k = 3$. If the number of neighbors in each majority sample is predominant from minority class samples, these majority class samples are removed as the overlapping samples. Another method of data cleaning is the neighborhood cleaning rule (NCL) for the removal of samples from the majority class [29].

Moreover, the noisy minority sample filter method was also combined with several methods of resampling as the pre-processing step [25]. And, the effectiveness of the minority noise filter method is examined in the imbalanced data classification. The class overlapping and imbalanced problems [9] are handled by combining with neighbor cleaning rules (NCL) and ensemble-based random under-sampling (ERUS) methods [29], [31]. The combination of Tomek Link and k-means clustering method [41] was also used as an effective under-sampling technique for addressing the class imbalance and overlapping problems.

In summary, many heuristic techniques have been conducted to improve the efficiency of imbalanced data classification. Most of the proposed methods would address the imbalanced problem among different classes, while some methods solved the class imbalanced and overlapping problems. Additionally, the problems of class imbalance and overlapping have been combated, but the three characteristics of skewed distribution problems have not been handled.

## III. Proposed Methodology

The main objective of this research work is to find an efficient way for the classification of imbalanced data. Fig. 2 shows the architecture overview of the proposed framework.

### A. Data Pre-processing Phase

In this paper, an effective resampling approach is proposed to improve the efficiency of classification algorithms, in which the class distribution of an imbalanced data set is changed before applying the learning algorithms.

The proposed approach combines the cluster-based resampling methods with Tomek Link method in order to alleviate the skewed distribution of data points from imbalanced data sets (denoted as COTU). The proposed COTU approach has been combined with the following three steps:

1) Cluster-Based Over-Sampling Step
2) Tomek Link-Based Under-Sampling Step
3) Cluster-Based Under-Sampling Step

Algorithm 1, 2 and 3 provide a brief description of proposed COTU approach.

**Algorithm 1** : Cluster-Based Over-Sampling Step

**Input:** Imbalanced Training Set, T = {*Min*, *Maj*}.
**Output:** New Set, $T'$ = {$Min'$, *Maj*}.

$K \leftarrow$ no: of clusters;
*Min* $\leftarrow$ set of minority samples in $T$;
*Maj* $\leftarrow$ set of majority samples in $T$;
$Min' \leftarrow \phi$;
$T' \leftarrow \phi$;
**procedure** SMALLDISJUNCTS ($T$, $K$)
  $Size_{Min} \leftarrow$ size of minority samples in $T$;
  $C \leftarrow$ k-means($Min$, $K$); **//Clustering**
  **for each** $c_i \in C$ **do**
    $Min^i \leftarrow$ set of minority samples in $c_i$;
    $Size^i_{Min} \leftarrow$ size of minority samples in $c_i$;
    **if** $Size^i_{Min} > 1$ **then**
      $Syn^i_{Min} \leftarrow \frac{Size_{Min}}{Size^i_{Min}}$;
      **if** $Syn^i_{Min} > Size^i_{Min}$ **then**
        $O \leftarrow$ over-sample, $Min^i$ into $Syn^i_{Min}$;
        $Min' \leftarrow Min' \cup \{O\}$;
      **else**
        $Min' \leftarrow Min' \cup \{Min^i\}$;
      **end if**
    **end if**
  **end for**
  $T' \leftarrow \{Min' \cup Maj\}$;
  **return** $T'$;
**end procedure**

TABLE I
CLUSTER-BASED OVER-SAMPLING DESCRIPTION

| $ClusterID$ | $Size_{Min}$ | $Syn^i_{Min}$ |
|---|---|---|
| 0 | 9 | 9 |
| 1 | 4 | 11 |
| 2 | 24 | 24 |
| 3 | 3 | 15 |
| 4 | 5 | 9 |
| $Size_{Min}$ = 45 | | $Size_{Min}$ = 68 |
| **Before Over-sampling** | | **After Over-sampling** |

*1) Cluster-Based Over-Sampling:* Class imbalance includes between-class imbalance and within-class imbalance [1], [39]. To achieve better prediction in imbalanced learning, the small disjuncts problem is one of the harmful problems. The small disjuncts problem relates to the within-class imbalance problem because the minority class is composed of several sub-concepts and the sizes of these sub-concepts are distinct, indicating a significant reduction in the efficiency of classification [24], [42].

In this paper, the proposed cluster-based over-sampling step, Algorithm (1) represents the first step in COTU approach to solving the small disjuncts problem, depending on the clustering of k-means and SMOTE [8].

In this Algorithm (1), the entire minority class is clustered into several groups and determines the sufficient number of synthesized samples ($Syn^i_{Min}$) in the $i^{th}$ cluster ($1 \leq i \leq K$) by considering the ratio of the size of minority class ($Size_{Min}$) to the number of minority class samples in the $i^{th}$ cluster ($Size^i_{Min}$). After identifying the number of minority samples in the $i^{th}$ cluster by using Eq. (1), we use SMOTE to over-sample the minority class samples.

**Algorithm 2** : Tomek Link-Based Under-Sampling Step

**Input:** Data Set, $T'$ = {$Min'$, $Maj$}.
**Output:** New Set, $T''$ = {$Min'$, $Maj'$}.

$Min' \leftarrow$ set of minority samples in $T'$;
*Maj* $\leftarrow$ set of majority samples in $T'$;
$neg \leftarrow$ majority class value in $T'$;
$Maj' \leftarrow \phi$;
$T'' \leftarrow \phi$;
**procedure** CLASSOVERLAPPING ($T'$)
  **for each** $x_i \in T'$ **do**
    $N \leftarrow 1^{st}$ nearest neighbor sample of $x_i$;
    **if** class($x_i$) ! = class($N$) **then**
      **if** class($x_i$) == "$neg$" **then**
        $X \leftarrow X \cup \{x_i\}$;
      **else**
        $X \leftarrow X \cup \{N\}$;
      **end if**
    **end if**
  **end for**
  $Maj' \leftarrow Maj - X$;
  $T'' \leftarrow \{Min' \cup Maj'\}$;
  **return** $T''$;
**end procedure**

For each cluster $i^{th}$,

$$Syn^i_{Min} = \frac{Size_{Min}}{Size^i_{Min}} \tag{1}$$

For example, Table I shows the number of synthesized minority samples in the $i^{th}$ cluster by calculating Eq. (1). Where, $Syn^i_{Min}$ is lower than $Size^i_{Min}$, $Size^i_{Min}$ acted as $Syn^i_{Min}$. After this step has been taken, the problem of small disjuncts is resolved in the training set.

*2) Tomek Link-Based Under-Sampling Step:* Tomek Link-Based Under-Sampling [43] is the second step of COTU approach, which is used to identify and eliminate the majority samples from the overlapping regions. The under-sampling of overlapping samples is also a reasonable approach for improving the performance of the learning algorithms [15], [45].

Tomek Link is a popular under-sampling method for solving the class overlapping problem, which based on two closely related samples from the different classes [11], [41]. In this paper, Algorithm (2) is a cleaning step for overlapping samples by reducing the majority samples, which are closely related to the minority samples.

*3) Cluster-Based Under-Sampling Step:* Another class imbalance problem is the between-class imbalance problem. The between-class imbalance problem corresponds to the small sample size problem, in which the size of minority class samples is smaller than the size of the majority class samples [39]. In the imbalanced learning, the prediction of learning classifier tends to favor the majority samples and ignores the minority samples, since the layout of learning classifiers is designed to provide the balanced data sets [21].

In this paper, the proposed cluster-based under-sampling step, Algorithm (3) is applied in COTU approach as the final

**Algorithm 3** Cluster-Based Under-Sampling Step

**Input:** Data Set, $T''$ = $\{Min', Maj'\}$.
**Output:** Balanced Data Set, $T'''$ = $\{Min', Maj''\}$.

$K \leftarrow$no: of clusters;
$Min' \leftarrow$ set of minority samples in $T''$;
$Maj' \leftarrow$ set of majority samples in $T''$;
$Maj'' \leftarrow \phi$;
$T''' \leftarrow \phi$;
**procedure** SMALLSAMPLESIZE ($T''$, $K$)
  $C \leftarrow$ k-means($Maj'$, $K$); // **Clustering**
  **for each** $c_i \in C$ **do**
    $Maj'^i \leftarrow$ set of majority samples in $c_i$;
    $Size^i_{Maj'} \leftarrow$ size of majority samples in $c_i$;
    $Size_{Maj'} \leftarrow$size of majority samples in $T''$;
    **if** $Size^i_{Maj'} > 1$ **then**
      $Und^i_{Maj} \leftarrow \frac{Size^i_{Maj'}}{Size_{Maj'}} * \sum_i^K Syn^i_{Min}$;
      **if** $Und^i_{Maj} < Size^i_{Maj'}$ **then**
        $R \leftarrow$ under-sample, $Maj'^i$ into $Und^i_{Maj}$;
        $Maj'' \leftarrow Maj'' \cup \{R\}$;
      **else**
        $Maj'' \leftarrow Maj'' \cup \{Maj'^i\}$;
      **end if**
    **end if**
  **end for**
  Balanced Data Set, $T''' \leftarrow \{Min' \cup Maj''\}$;
  **return** $T'''$;
**end procedure**

TABLE II
: CLUSTER-BASED UNDER-SAMPLING DESCRIPTION

| $ClusterID$ | $Size^i_{Maj}$ | $Und^i_{Maj}$ |
|---|---|---|
| 0 | 17 | 14 |
| 1 | 3 | 2 |
| 2 | 39 | 31 |
| 3 | 22 | 18 |
| 4 | 4 | 3 |
| $Size_{Maj}$ = 85 | | $Size_{Maj}$ = 68 |
| **Before Under-sampling** | | **After Under-sampling** |

step for solving the small sample size problem, depending on k-means clustering and RUS methods [3]. In this Algorithm (3), the whole majority class samples set into some clusters, and determines the number of selected majority samples ($Und^i_{Maj}$) for the $i^{th}$ cluster (1 $\leq$ i $\leq$ $K$) by Eq. (2). After determining the number of selected majority samples, the majority samples in each cluster are randomly selected [3].

For each cluster $i^{th}$,

$$Und^i_{Maj} = \frac{Size^i_{Maj}}{Size_{Maj}} * \sum_i^K Syn^i_{Min} \quad (2)$$

In Eq. (2), the number of selected majority samples for the $i^{th}$ cluster ($Und^i_{Maj}$) is the combination of $\sum_i^K Syn^i_{Min}$ which is the total number of synthesized samples that are supported with Eq. (1) and ($\frac{Size^i_{Maj}}{Size_{Maj}}$) which is the ratio of the number of majority samples in the $i^{th}$ cluster to the number of whole majority class samples. Table II describes the example of a cluster-based under-sampling step for handling the small sample size problem.

After COTU approach has been established, a new balanced training data set can be created by avoiding the three challenges of skewed distribution.

### *B. Classification Phase*

After executing the data pre-processing phase, the refining training data set is fed into the classification phase to predict the unknown input variables for a specific class. In this paper, Decision Tree (C4.5) and Navies Bayes (NB) are used to investigate the impact of COTU approach and existing data pre-processing techniques on classification tasks by using the WEKA Machine Learning Tool, which is an open-source data mining suite [20]. C4.5 is a learner of a decision tree using an entropy-based splitting criterion based on information theory [21], [37]. In this study, we used J48 algorithm in Weka to implement C4.5. Naive Bayes (NB) uses Bayes's rule of conditional probability to classify samples, where all variables of the predictor are conditionally independent [21].

Furthermore, the individual classifier is combined with AdaBoost mechanism (such as AdaBoost-C4.5 and AdaBoost-NB) to form the final prediction model. AdaBoost is a popular boosting algorithm that uses the entire data set to serially train each classification, but it focuses on misclassifying samples in every round to correctly classify samples in the next iteration [16], [17]. In our evaluation, all learning classifiers are constructed with the default parameters setting of WEKA Machine Learning Tool [20].

### *C. Result Analysis Phase*

In this section, the details of the results analysis arrangements for evaluating the performance of COTU approach are presented. First, we described the performance metrics for experimental analysis. Then, the accuracy of the probabilistic estimation is briefly described. Moreover, Wilcoxon Signed Rank Test is discussed to perform a statistical analysis of all experiments. Finally, TOPSIS strategy is discussed to decide the best sampling method for each learning classifier.

*1) Performance Metrics:* The performance of learning algorithms is evaluated by the confusion matrix. The confusion matrix of a two-class problem records the results correctly and incorrectly recognized samples of each class [25]. This research work focuses on the two-class imbalanced data sets, where the smallest number of instances is a positive class (minority class) and the largest number of instances is a negative class (majority class).

In the class imbalance learning, the performance measures are crucial to evaluate the performance learning algorithm. Some widely used measures, such as accuracy, the predominance of majority class can mask the poor classification performance for minority class. The imbalanced data performance metrics must depend on results from both classes, and measurements for each class are inadequate [41]. The main performance metrics of this research are Area Under the Curve (AUC) Eq. (3) Geometric Mean (G-mean) Eq. (4) and Balance Accuracy (BA) Eq. (5). AUC, G-mean and BA metrics are evaluated based on the results of the majority class and the minority class, simultaneously.

In Eq. (3) to (5), TP is the number of minority samples correctly predicted (True Positive), TN is the number of majority samples correctly predicted (True Negatives) and FP is the number of majority samples incorrectly predicted as minority samples (False Positives) [35].

$$AUC = \frac{(1 + TPrate - FPrate)}{2} \tag{3}$$

$$G - mean = \sqrt{(TPrate \times TNrate)} \tag{4}$$

$$BA = \frac{TPrate + TNrate}{2} \tag{5}$$

Area under the ROC curve (AUC) represents a trade-off measure between TP rates and FP rates. It estimates the area under the receiver operating characteristics (ROC) curves. ROC is obtained by plotting a set of TP rates versus FP rates related to various classification thresholds [14]. A higher AUC shows better classifier efficiency [9], [33]. G-mean is the geometric average of TP rate and TN rate. G-mean value produces the high scores when errors in the majority and minority classes are low [28]. Similarly, BA is the arithmetic average of true positive and true negative rates [44], as shown in Eq. (5). The measurement of BA is identical to the conventional accuracy when the classifier performs equally well in either class. However, if the conventional accuracy is high only because the classifier takes advantage of good prediction on the majority class, then the balanced accuracy will be lower [4], [6].

So, AUC, G-mean and BA are the best metrics for evaluating model performances, as these measures can provide an opportunity to evaluate the results of majority class and minority class, simultaneously [2], [41].

*2) Accuracy of Probabilistic Estimation:* To estimate the expected costs of classification decisions, Brier score (BS) Eq. (6) [5] [47] is used. BS is a measure of the average squared difference between the estimated probabilities and the observed class value. $(y_i - \hat{P}y_i|x_i)$ is the difference between the observed labels and the corresponding probability estimation. In Eq. (6), we are assuming that, y $\in \{0,1\}$ and $N$ is the number of instances in the test set. For the probabilistic predictions of a model, the lower brier score is more accurate.

$$BS = \frac{\sum_{i=1}^{N}(y_i - \hat{P}y_i|x_i)^2}{N} \tag{6}$$

However, when estimating the probability of imbalanced scenario, the overall Brier score (BS) may increase the uncertainty due to the underestimation of the probability of minority class instances. So, the stratified Brier score is used to measure the probability calibration Eq. (7) and Eq. (8) offer a lot more detail than the overall Brier score (BS) Eq. (6) because the stratified Brier score can provide the model calibration for both classes [47]. $BS^+$ is a Brier score of minority class which measures the goodness of the minority class estimation. In Eq. (7), $\hat{P}y_i|x_i$ is the probability estimate of minority sample $x_i$ and $N_{min}$ is the total number minority class samples. And also, $BS^-$ Eq. (8) is a Brier score for the best estimation of the majority class, in which $\hat{P}y_i|x_i$ is the probability estimate of majority sample $x_i$ and $N_{maj}$ is the total number majority class samples.

$$BS^+ = \frac{\sum_{y_i=1}^{N_{min}}(y_i - \hat{P}y_i|x_i)^2}{N_{min}} \tag{7}$$

$$BS^- = \frac{\sum_{y_i=1}^{N_{maj}}(y_i - \hat{P}y_i|x_i)^2}{N_{maj}} \tag{8}$$

$$BS^{Bal} = \frac{BS^+ + BS^-}{2} \tag{9}$$

$BS^{Bal}$ Eq. (9) is the arithmetic mean of $BS^+$ and $BS^-$, which measures the probability estimations of minority class and majority class simultaneously. If $BS^{Bal}$ is good, the probability estimation of minority class and majority is completely reliable.

*3) Statistical Significance Tests:* In our Statistical Significance Tests, Wilcoxon Signed Rank Test is performed with SPSS software to test the statistical analysis of comparative results. Wilcoxon's Signed Rank Test was a hypothetical test of non-parametric statistical tests that used to compare two different sampling methods for the same data sets and assess for the significant difference (i.e. It is a paired difference test) [12], [48].

In Wilcoxon's Signed Rank Test, $W^+$ and $W^-$ are the sum ranks of positive and negative, and $P_{Wilcoxon}$ is the $P_{value}$ of Wilcoxon's test. $Win\text{-}Tie\text{-}Loss$ is the number of cases, where the results of COTU approach are $greater - than$, $equal - to$ and $less - than$ the comparison methods. For all statistical tests of this paper, the level of significance is set at $P_{value} = 0.05$. When $P_{Wilcoxon}$ is less than 0.05, the comparison of the paired difference methods will be significantly different. In our statistical analysis, all significant differences in each data pre-processing techniques are highlighted in boldface.

*4) Sampling Method Selection:* In this part, we illustrate how to determine the appropriate sampling method for each learning model using TOPSIS strategy in the MCDM methodology. TOPSIS is the best way to select the best action from a set of alternatives, each of which is evaluated using several criteria. TOPSIS finds the best possible alternative by minimizing the distance from the positive ideal solution (PIS) and maximizing the distance from the negative ideal solution (NIS) [38], [40].

In the experiment, the goodness of sampling methods is measured based on AUC, G-mean, BA and $BS^{Bal}$ results from 25 imbalanced data sets (Criteria) and six sampling methods (Alternatives). Then, the results of TOPSIS are collected and detailed analyses are carried out in the section of results and discussions.

The processes of TOPSIS strategy for the experimental analysis is summarized as follows:

Where, the construction of a decision matrix (DM) is the first step of TOPSIS strategy. In Table III, $i$ is the alternative

TABLE III
DECISION MATRIX CONSTRUCTION

| *Alternatives* | *Criteria* | | | | |
|---|---|---|---|---|---|
| | $C_1$ | $C_2$ | $C_3$ | ... | $C_y$ |
| $A_1$ | $a_{11}$ | $a_{12}$ | $a_{13}$ | ... | $a_{1y}$ |
| $A_2$ | $a_{21}$ | $a_{22}$ | $a_{23}$ | ... | $a_{2y}$ |
| : | : | : | : | | : |
| $A_x$ | $a_{x1}$ | $a_{x2}$ | $a_{x3}$ | ... | $a_{xy}$ |

index $(i = 1, ..., x)$ and $j$ is the criteria index $(j = 1, ..., y)$. The elements $C_1$, $C_2$, ..., $C_y$ refer to the criteria, while $A_1$, $A_2$,..., $A_x$ refer to the alternatives. Alternative $A_i$'s performance value on the criterion $C_j$ is represented by $a_{ij}$.

**Step 1 :** Calculate the normalized decision matrix. The normalized value $r_{ij}$ is calculated as;

$$r_{ij} = \frac{a_{ij}}{\sqrt{\sum a_{ij}^2}} \tag{10}$$

**Step 2 :** The weighted normalized decision matrix is constructed while the criterion's weight value plays an important role for MCDM technique, which affects the final ranking of alternatives.

In the experiments of this paper, the criterion weights are determined by ($W_j$=$\frac{1}{n}$). The weighted normalized value $v_{ij}$ is calculated as $V$.

$$V = v_{ij} = W_j * r_{ij} \tag{11}$$

Where $W_j$ is the weight of the $j^{th}$ criterion and $n$ is the number of criteria.

$$\sum_{j=1}^{K} W_j = 1 \tag{12}$$

**Step 3 :** Determine the positive ideal ($A^+$) and negative ideal solutions ($A^-$) that are defined by the weighted decision matrix.

(1) Positive Ideal Solution
PIS = $A^+$ = $V_1^+$, $V_2^+$, $V_3^-$,..., $V_n^+$, where:
$V_j^+$ = max $(v_{ij})$ if j ∈ J; min$(v_{ij})$ if j ∈ $J^{'}$

(2) Negative Ideal Solution
NIS = $A^-$ = $V_1^-$, $V_2^-$, $V_3^-$,..., $V_n^-$, where:
$V_j^-$ = min $(v_{ij})$ if j ∈ J; max$(v_{ij})$ if j ∈ $J^{'}$

Where, $J$ is associated with the beneficial attributes and $J'$ is associated with the non-beneficial attributes.

**Step 4 :** Calculate the distance between positive and negative for each competitive alternative:
(1) Positive Ideal Separation

$$S^+ = \sqrt{\sum_{j=1}^{y} \left(V_j^+ - v_{ij}\right)^2} \tag{13}$$

(2) Negative Ideal Separation

$$S^- = \sqrt{\sum_{j=1}^{y} \left(V_j^- - v_{ij}\right)^2} \tag{14}$$

TABLE IV
CHARACTERISTICS OF EXPERIMENTAL DATA SETS

| *DataSets* | $\#Att$ | $\#Maj$ | $\#Min$ | $IR$ |
|---|---|---|---|---|
| glass1 | 9 | 138 | 76 | 1.82 |
| vehicle1 | 18 | 629 | 217 | 2.9 |
| vehicle3 | 18 | 634 | 212 | 2.99 |
| vehicle0 | 18 | 647 | 199 | 3.25 |
| new-thyroid1 | 5 | 180 | 35 | 5.14 |
| newthyroid2 | 5 | 180 | 35 | 5.14 |
| segment0 | 19 | 1979 | 329 | 6.02 |
| glass6 | 9 | 185 | 29 | 6.38 |
| yeast3 | 8 | 1321 | 163 | 8.1 |
| ecoli-034_Vs_5 | 7 | 180 | 20 | 9 |
| yeast-2_Vs_4 | 8 | 463 | 51 | 9.08 |
| yeast-02579_Vs_368 | 8 | 905 | 99 | 9.14 |
| ecoli-0267_Vs_35 | 7 | 202 | 22 | 9.18 |
| ecoli-067_Vs_5 | 6 | 200 | 20 | 10 |
| led7digit-02456789_Vs_1 | 7 | 406 | 37 | 10.97 |
| glass-0146_Vs_2 | 9 | 188 | 17 | 11.06 |
| glass2 | 9 | 197 | 17 | 11.59 |
| ecoli4 | 7 | 316 | 20 | 15.8 |
| glass-016_Vs_5 | 9 | 175 | 9 | 19.44 |
| yeast-2_Vs_8 | 8 | 462 | 20 | 23.1 |
| yeast4 | 8 | 1240 | 244 | 28.1 |
| winequality-white-9_Vs_4 | 11 | 163 | 5 | 32.6 |
| ecoli-0137_Vs_26 | 7 | 274 | 7 | 39.14 |
| yeast6 | 8 | 1449 | 35 | 41.4 |
| winequality-white-39_Vs_5 | 11 | 1457 | 25 | 58.28 |

**Step 5 :** For ideal solution, calculate the relative closeness.

$$C_i = \frac{S_i^-}{(S_i^+ + S_i^-)} \tag{15}$$

**Step 6 :** Finally, rank the order of preferences.

## IV. EXPERIMENTAL SETTINGS

This section presents the experimental arrangements for evaluating the performance of COTU approach. The experimental setting has the two things. First, we demonstrate the nature of training data sets. Then, some sampling methods are discussed to evaluate and compare the performance of COTU approach.

### A. Data Sets

In order to provide conclusive results, we use 25 imbalanced data sets that are taken from the KEEL-Data Set Repository [1]. Table IV provides the main characteristics of these data sets. For each one, the number of attributes ($\#Att$), the number of majority class samples ($\#Maj$), the number of minority class samples ($\#Min$) and the ratio of majority class samples to minority ones ($IR$) [25] are shown.

These data sets have $IR$ ratios between $1.82$ and $58.28$, with the number of samples ranging from $168$ samples ($winequality - white - 9\_vs\_4$) to $2308$ samples ($segment0$). And, the number of attributes is ranging from $5$ to $19$ attributes. In Table IV, the data sets are sorted from the lowest $IR$ to the highest $IR$ ratio.

[1]htt,http://sci2s.ugr.es/keel/imbalanced.php

TABLE V
AUC RESULTS FOR 25 DATA SETS AND SIX DATA PRE-PROCESSING TECHNIQUES USING ADABOOST-C4.5 AND ADABOOST-NB CLASSIFIERS

| Data Sets | AdaBoost-C4.5 | | | | | | AdaBoost-NB | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | SMOTE | Center - NN | SBC | BDSK | SMOTE - TL | COTU | SMOTE | Center - NN | SBC | BDSK | SMOTE - TL | COTU |
| glass1 | **0.875** | 0.853 | 0.745 | 0.845 | 0.867 | **0.875** | 0.732 | **0.753** | 0.721 | 0.703 | 0.731 | 0.729 |
| vehicle1 | 0.83 | 0.826 | 0.806 | 0.835 | 0.826 | **0.841** | 0.778 | 0.764 | 0.768 | 0.737 | **0.779** | 0.773 |
| vehicle3 | **0.827** | 0.825 | 0.81 | 0.804 | 0.823 | 0.823 | 0.765 | 0.77 | 0.764 | 0.743 | **0.778** | **0.778** |
| vehicle0 | **0.994** | 0.988 | 0.764 | 0.991 | 0.987 | 0.986 | **0.963** | 0.926 | 0.795 | 0.942 | 0.958 | 0.962 |
| new-thyroid1 | 0.962 | 0.961 | 0.965 | 0.961 | 0.977 | **0.996** | **1** | 0.997 | 0.997 | 0.996 | **1** | 0.996 |
| newthyroid2 | 0.956 | 0.961 | 0.944 | 0.957 | **0.966** | 0.958 | 0.998 | 0.997 | 0.998 | **0.999** | 0.998 | 0.998 |
| segment0 | 0.997 | 0.995 | 0.951 | **0.998** | 0.995 | 0.995 | 0.996 | 0.98 | 0.951 | **0.998** | 0.996 | 0.985 |
| glass6 | 0.944 | 0.959 | 0.937 | 0.935 | 0.944 | **0.964** | 0.885 | 0.899 | 0.948 | 0.873 | **0.96** | 0.949 |
| yeast3 | 0.956 | 0.956 | 0.957 | 0.952 | 0.955 | **0.961** | 0.951 | 0.95 | 0.942 | 0.945 | **0.963** | 0.958 |
| ecoli-034_Vs_5 | 0.96 | 0.96 | 0.958 | **0.974** | 0.958 | 0.963 | 0.947 | 0.937 | 0.923 | **0.952** | 0.897 | 0.921 |
| yeast-2_Vs_4 | **0.974** | 0.957 | 0.94 | 0.958 | 0.966 | 0.971 | 0.961 | 0.947 | 0.935 | 0.928 | 0.96 | **0.969** |
| yeast-02579_Vs_368 | **0.954** | 0.946 | 0.943 | 0.932 | 0.95 | 0.952 | 0.909 | 0.91 | 0.893 | 0.856 | 0.905 | **0.914** |
| ecoli-0267_Vs_35 | 0.906 | **0.911** | 0.888 | 0.855 | 0.857 | 0.896 | 0.903 | 0.87 | 0.914 | 0.878 | **0.923** | 0.856 |
| ecoli-067_Vs_5 | 0.901 | 0.897 | 0.941 | **0.956** | 0.931 | 0.943 | 0.855 | 0.843 | **0.876** | 0.831 | 0.848 | 0.872 |
| led7digit-02456789_Vs_1 | 0.955 | **0.961** | 0.92 | 0.939 | 0.906 | 0.948 | 0.919 | 0.908 | 0.904 | 0.933 | 0.904 | **0.934** |
| glass-0146_Vs_2 | 0.849 | 0.73 | 0.767 | 0.812 | **0.93** | 0.845 | 0.611 | 0.652 | 0.723 | 0.721 | 0.607 | **0.736** |
| glass2 | **0.905** | 0.565 | 0.539 | 0.845 | 0.858 | 0.866 | 0.638 | 0.665 | 0.649 | 0.733 | 0.634 | **0.744** |
| ecoli4 | 0.914 | 0.931 | 0.86 | **0.966** | 0.911 | 0.95 | 0.943 | 0.99 | **0.995** | 0.959 | 0.99 | 0.987 |
| glass-016_Vs_5 | 0.894 | 0.943 | 0.943 | **0.974** | 0.891 | 0.959 | 0.94 | **0.971** | 0.866 | 0.967 | 0.94 | 0.946 |
| yeast-2_Vs_8 | 0.776 | 0.785 | 0.8 | 0.845 | **0.875** | 0.83 | 0.771 | 0.723 | 0.78 | **0.795** | 0.75 | 0.792 |
| yeast4 | 0.899 | 0.872 | 0.892 | 0.898 | **0.904** | 0.902 | 0.885 | 0.858 | 0.88 | 0.821 | **0.887** | 0.876 |
| winequality-white-9_Vs_4 | 0.788 | 0.814 | 0.639 | **0.891** | 0.794 | 0.825 | 0.578 | **0.836** | 0.571 | 0.682 | 0.669 | 0.785 |
| ecoli-0137_Vs_26 | 0.843 | 0.807 | 0.773 | 0.826 | 0.834 | **0.872** | 0.878 | **0.974** | 0.889 | 0.888 | 0.828 | 0.967 |
| yeast6 | 0.898 | 0.903 | 0.918 | 0.901 | 0.928 | **0.93** | 0.925 | 0.906 | 0.894 | 0.876 | 0.912 | **0.928** |
| winequality-white-39_Vs_5 | 0.669 | 0.633 | 0.654 | 0.638 | **0.74** | 0.674 | 0.76 | 0.745 | 0.718 | 0.683 | 0.75 | **0.778** |
| **Avg. Val** | 0.897 | 0.878 | 0.850 | 0.900 | 0.903 | **0.909** | 0.860 | 0.871 | 0.852 | 0.858 | 0.863 | **0.885** |
| **Avg. Rank** | 3 | 3.9 | 4.9 | 3.6 | 3.4 | **2.2** | 3.2 | 3.6 | 4.1 | 4.1 | 3.3 | **2.6** |

### *B. Compared Methods*

In our experiments, the performance of COTU approach has been compared with the five state-of-the-art sampling methods. Where, SMOTE is an over-sampling method and Center-NN and SBC are the under-sampling methods. BDSK, SMOTE-TL and the proposed COTU approach are hybrid resampling methods. In COTU approach, the setting of cluster numbers for small disjuncts and small sample size problems is set to five.

All pre-processing methods are implemented in the framework of the WEKA Machine Learning Tool and used the same classification algorithms for each set of experiments. The details of comparative resampling methods are provided as follows:

*1) Synthetic Minority Over-Sampling Technique (SMOTE):* SMOTE [8] is an over-sampling method that generates the synthetic samples based on the minority class samples and its $k^{th}$ nearest neighbors. To generate new samples, the $k^{th}$ nearest neighbor samples are selected based on a user-defined threshold ($k$). In our evaluation, for all SMOTE-related procedures, $k = 5$ is assigned to maintain the same parameter settings for all methods and data sets.

To generate the synthesis samples, SMOTE operates with three steps. Firstly, a minority sample ($a$) is randomly selected. And, its $k^{th}$ nearest minority class neighbor ($x$) is randomly selected. Finally, the new sample ($n$) is generated by the following formula Eq. (16):

$$n = a + r * (x - a) \tag{16}$$

Where, ($r$) is the random weight between 0 and 1. SMOTE method is the available jar file that is created by Chawla et al., [8].

*2) Center-NN:* The cluster-based under-sampling method (Center-NN) [30] uses k-means clustering method to cluster all majority samples into multiple clusters. In the Center-NN method, the number of clusters is equal to the sample size of minority class. Then, the nearest samples of all cluster centers are selected as the majority class. The selected majority samples and minority samples are combined as the new training set for applying the learning algorithms. The target of Center-NN is to solve the between-class imbalanced problem.

*3) SBC:* SBC is also the cluster-based under-sampling method for solving the between-class imbalanced problem, [50] in which all training samples are divided into $k$ clusters and the subset of majority samples from each cluster is randomly selected, based on the ratios of majority samples to minority samples. Finally, the selected majority samples and the entire minority class samples are combined as the

TABLE VI
G-MEAN RESULTS FOR 25 DATA SETS AND SIX DATA PRE-PROCESSING TECHNIQUES USING ADABOOST-C4.5 AND ADABOOST-NB CLASSIFIERS

| Data Sets | AdaBoost-C4.5 | | | | | | AdaBoost-NB | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | SMOTE | Center - NN | SBC | BDSK | SMOTE - TL | COTU | SMOTE | Center - NN | SBC | BDSK | SMOTE - TL | COTU |
| glass1 | 0.803 | 0.79 | 0.622 | 0.773 | 0.75 | **0.813** | 0.625 | 0.631 | 0.585 | 0.678 | 0.654 | **0.695** |
| vehicle1 | 0.706 | 0.73 | 0.72 | 0.748 | 0.734 | **0.764** | 0.696 | 0.71 | 0.682 | 0.663 | **0.717** | 0.699 |
| vehicle3 | 0.702 | 0.74 | 0.745 | 0.704 | 0.746 | **0.76** | 0.704 | 0.705 | 0.703 | 0.691 | 0.714 | **0.725** |
| vehicle0 | 0.96 | 0.946 | 0.727 | **0.961** | 0.956 | 0.938 | 0.886 | 0.865 | 0.741 | 0.851 | 0.879 | **0.889** |
| new-thyroid1 | 0.949 | 0.96 | 0.945 | 0.934 | **0.974** | **0.974** | 0.98 | 0.977 | 0.954 | 0.962 | **0.989** | 0.977 |
| newthyroid2 | 0.938 | 0.945 | 0.906 | 0.953 | 0.95 | **0.954** | 0.977 | 0.969 | 0.969 | **0.992** | 0.971 | 0.983 |
| segment0 | **0.995** | 0.991 | 0.899 | 0.994 | 0.991 | 0.991 | 0.979 | 0.938 | 0.868 | **0.987** | 0.974 | 0.961 |
| glass6 | 0.902 | 0.913 | 0.894 | 0.896 | 0.909 | **0.916** | 0.847 | 0.867 | 0.931 | 0.855 | 0.881 | **0.939** |
| yeast3 | 0.867 | 0.902 | 0.889 | 0.885 | **0.924** | 0.917 | 0.881 | 0.898 | 0.881 | 0.87 | **0.914** | 0.904 |
| ecoli-034_Vs_5 | **0.936** | 0.893 | 0.87 | 0.907 | 0.891 | 0.87 | 0.878 | 0.857 | **0.881** | 0.872 | 0.876 | 0.867 |
| yeast-2_Vs_4 | 0.872 | 0.911 | **0.913** | 0.884 | **0.913** | **0.913** | 0.853 | 0.879 | 0.865 | 0.804 | 0.854 | **0.906** |
| yeast-02579_Vs_368 | 0.895 | 0.875 | 0.868 | 0.882 | **0.905** | 0.891 | 0.895 | 0.865 | 0.831 | 0.766 | 0.887 | **0.899** |
| ecoli-0267_Vs_35 | 0.836 | 0.818 | 0.787 | 0.846 | 0.835 | **0.856** | **0.87** | 0.823 | 0.777 | 0.551 | 0.853 | 0.797 |
| ecoli-067_Vs_5 | 0.809 | 0.817 | 0.865 | **0.919** | 0.809 | 0.868 | **0.823** | 0.769 | 0.821 | 0.822 | **0.823** | 0.796 |
| led7digit-02456789_Vs_1 | **0.9** | 0.869 | 0.862 | 0.873 | 0.515 | 0.885 | 0.851 | **0.89** | 0.85 | 0.804 | 0.868 | 0.875 |
| glass-0146_Vs_2 | 0.549 | 0.578 | 0.624 | 0.437 | **0.84** | 0.743 | 0.559 | 0.564 | 0.59 | 0.531 | 0.559 | **0.653** |
| glass2 | 0.651 | 0.536 | 0.513 | 0.521 | 0.545 | **0.824** | 0.603 | 0.546 | 0.581 | 0.498 | 0.609 | **0.656** |
| ecoli4 | 0.853 | 0.913 | 0.842 | **0.965** | 0.88 | 0.902 | 0.884 | 0.925 | **0.951** | 0.909 | 0.886 | 0.941 |
| glass-016_Vs_5 | 0.794 | 0.941 | 0.941 | 0.934 | 0.794 | **0.959** | **0.794** | 0.906 | 0.864 | 0.536 | 0.791 | 0.891 |
| yeast-2_Vs_8 | 0.612 | 0.718 | 0.724 | 0.76 | **0.793** | 0.761 | 0.693 | 0.688 | 0.636 | 0.678 | 0.695 | **0.727** |
| yeast4 | 0.653 | 0.776 | **0.822** | 0.678 | 0.742 | 0.809 | **0.794** | 0.759 | 0.783 | 0.703 | 0.755 | 0.781 |
| winequality-white-9_Vs_4 | 0.587 | 0.728 | 0.393 | 0.597 | 0.584 | **0.789** | 0.397 | 0.708 | 0.198 | 0.397 | 0.397 | **0.713** |
| ecoli-0137_Vs_26 | 0.533 | 0.689 | 0.706 | 0.538 | **0.735** | 0.664 | 0.735 | 0.707 | 0.669 | 0.536 | 0.73 | **0.762** |
| yeast6 | 0.769 | 0.849 | 0.843 | 0.778 | 0.782 | **0.852** | 0.848 | 0.831 | 0.812 | 0.787 | 0.865 | **0.88** |
| winequality-white-39_Vs_5 | 0.349 | 0.568 | 0.587 | 0.175 | 0.358 | **0.602** | 0.678 | **0.722** | 0.657 | 0.307 | 0.639 | 0.667 |
| **Avg. Val** | 0.777 | 0.816 | 0.780 | 0.782 | 0.794 | **0.849** | 0.789 | 0.800 | 0.763 | 0.722 | 0.791 | **0.823** |
| **Avg. Rank** | 4.2 | 3.5 | 4.4 | 3.7 | 3.2 | **2** | 3.2 | 3.5 | 4.3 | 4.9 | 3 | **2.1** |

new training set for building the learning model.

*4) BDSK:* The hybrid method, BDSK [39] can handle within-class imbalanced and between-class imbalanced problems. In the under-sampling stage of BDSK, the majority class samples are clustered into the mean class size groups, and the nearest samples of cluster centers from each group [30] are chosen to solve the between-class imbalance problem.

Subsequently, the minority class samples are clustered into two clusters, and the smaller cluster is over-sampled by using SMOTE to deal with the within-class imbalance problem. In the over-sampling stage, SMOTE is performed until both classes are equal in size. Since, BDSK clusters both classes separately, is oblivious to the overlapping class and can lead the generation of noise.

*5) SMOTE-TL:* SMOTE in combination with Tomek Link (SMOTE-TL) [3] is a hybrid technique, through using SMOTE [8], and to remove noise and borderline samples by combining Tomek Link [43]. In SMOTE-TL, SMOTE over-samples the original data set, and removes the noisy and borderline samples by using Tomek Links. So, SMOTE-TL can solve the class imbalance and overlapping problems between two classes.

## V. EXPERIMENTAL RESULTS AND ANALYSIS

In this paper, the experiments are conducted with four learning classifiers (AdaBoost-C4.5, AdaBoost-NB, C4.5, and NB) by applying the proposed COTU approach and the five state-of-the-arts resampling techniques (SMOTE, Center-NN, SBC, BDSK, and SMOTE-TL) on 25 imbalanced data sets. The main aim of this section is to examine the impact of COTU approach. For all experimental evaluations, a 5-fold cross-validation approach [26] is used for the training set and testing set.

During the experimental analysis, the analysis of experimental results is divided into five parts, according to the evaluation criteria. The first four parts offer a detailed analysis of the effectiveness of the COTU approach over the ensemble-based learning classifiers. The last part is the overall results of four learning classifiers for all representative resampling methods. In all experiments, the test models achieving the highest results for each data set and all significant results are highlighted in boldface.

### *A. Classification Results by Ensemble-Based Learning Classifiers*

In the first experimental analysis, we contrasted our COTU approach with SMOTE, Center-NN, SBC, BDSK, and SMOTE-TL by comparing the class label prediction measures. Tables V to VII report the AUC, G-mean and BA results of each resampling method across the two learning

TABLE VII
BA RESULTS FOR 25 DATA SETS AND SIX DATA PRE-PROCESSING TECHNIQUES USING ADABOOST-C4.5 AND ADABOOST-NB CLASSIFIERS

| Data Sets | AdaBoost-C4.5 | | | | | | AdaBoost-NB | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | SMOTE | Center - NN | SBC | BDSK | SMOTE - TL | COTU | SMOTE | Center - NN | SBC | BDSK | SMOTE - TL | COTU |
| glass1 | 0.805 | 0.792 | 0.66 | 0.775 | 0.756 | **0.814** | 0.659 | 0.671 | 0.639 | 0.703 | 0.673 | **0.717** |
| vehicle1 | 0.721 | 0.731 | 0.739 | 0.752 | 0.736 | **0.766** | 0.699 | 0.712 | 0.704 | 0.667 | **0.721** | 0.706 |
| vehicle3 | 0.715 | 0.741 | 0.749 | 0.713 | 0.748 | **0.765** | 0.709 | 0.714 | 0.71 | 0.695 | 0.725 | **0.73** |
| vehicle0 | 0.96 | 0.947 | 0.764 | **0.961** | 0.956 | 0.939 | 0.887 | 0.868 | 0.775 | 0.852 | 0.881 | **0.889** |
| new-thyroid1 | 0.952 | 0.961 | 0.947 | 0.937 | **0.975** | **0.975** | 0.98 | 0.978 | 0.956 | 0.963 | **0.989** | 0.978 |
| newthyroid2 | 0.94 | 0.947 | 0.911 | 0.954 | 0.952 | **0.955** | 0.977 | 0.969 | 0.969 | **0.992** | 0.972 | 0.983 |
| segment0 | **0.995** | 0.991 | 0.904 | 0.994 | 0.991 | 0.991 | 0.979 | 0.94 | 0.877 | **0.987** | 0.974 | 0.962 |
| glass6 | 0.906 | 0.915 | 0.896 | 0.9 | 0.912 | **0.918** | 0.855 | 0.87 | 0.932 | 0.861 | 0.888 | **0.94** |
| yeast3 | 0.872 | 0.902 | 0.891 | 0.888 | **0.925** | 0.917 | 0.885 | 0.899 | 0.883 | 0.873 | **0.915** | 0.905 |
| ecoli-034_Vs_5 | **0.939** | 0.9 | 0.875 | 0.917 | 0.9 | 0.881 | **0.889** | 0.867 | **0.889** | 0.883 | 0.886 | 0.878 |
| yeast-2_Vs_4 | 0.877 | 0.914 | 0.915 | 0.889 | **0.916** | 0.914 | 0.856 | 0.879 | 0.867 | 0.815 | 0.859 | **0.908** |
| yeast-02579_Vs_368 | 0.899 | 0.875 | 0.87 | 0.887 | **0.907** | 0.892 | 0.897 | 0.867 | 0.834 | 0.777 | 0.889 | **0.899** |
| ecoli-0267_Vs_35 | 0.853 | 0.828 | 0.796 | **0.863** | 0.85 | **0.863** | **0.873** | 0.836 | 0.792 | 0.693 | 0.863 | 0.804 |
| ecoli-067_Vs_5 | 0.828 | 0.825 | 0.875 | **0.922** | 0.825 | 0.87 | 0.832 | 0.782 | 0.828 | 0.832 | **0.833** | 0.805 |
| led7digit-02456789_Vs_1 | **0.906** | 0.873 | 0.866 | 0.877 | 0.64 | 0.887 | 0.855 | **0.892** | 0.852 | 0.814 | 0.872 | 0.877 |
| glass-0146_Vs_2 | 0.674 | 0.602 | 0.671 | 0.635 | **0.847** | 0.756 | 0.59 | 0.62 | 0.664 | 0.59 | 0.59 | **0.711** |
| glass2 | 0.758 | 0.553 | 0.538 | 0.661 | 0.665 | **0.847** | 0.634 | 0.573 | 0.637 | 0.621 | 0.639 | **0.703** |
| ecoli4 | 0.867 | 0.915 | 0.847 | **0.967** | 0.892 | 0.904 | 0.891 | 0.928 | **0.953** | 0.914 | 0.892 | 0.942 |
| glass-016_Vs_5 | 0.894 | 0.943 | 0.943 | 0.941 | 0.891 | **0.96** | 0.894 | **0.916** | 0.874 | 0.741 | 0.891 | 0.901 |
| yeast-2_Vs_8 | 0.727 | 0.731 | 0.737 | 0.792 | **0.815** | 0.772 | 0.725 | 0.695 | 0.652 | 0.719 | 0.731 | **0.745** |
| yeast4 | 0.708 | 0.778 | **0.825** | 0.725 | 0.77 | 0.811 | **0.801** | 0.762 | 0.787 | 0.727 | 0.763 | 0.784 |
| winequality-white-9_Vs_4 | 0.788 | 0.814 | 0.639 | 0.791 | 0.784 | **0.825** | 0.691 | **0.805** | 0.534 | 0.697 | 0.694 | 0.801 |
| ecoli-0137_Vs_26 | 0.741 | 0.783 | 0.771 | 0.746 | **0.843** | 0.765 | 0.843 | 0.815 | 0.779 | 0.743 | 0.837 | **0.864** |
| yeast6 | 0.798 | 0.854 | 0.846 | 0.803 | 0.807 | **0.855** | 0.856 | 0.836 | 0.818 | 0.805 | 0.87 | **0.881** |
| winequality-white-39_Vs_5 | 0.558 | 0.581 | 0.612 | 0.524 | 0.587 | **0.639** | 0.704 | **0.734** | 0.677 | 0.535 | 0.685 | 0.693 |
| **Avg. Val** | 0.827 | 0.828 | 0.803 | 0.833 | 0.836 | **0.859** | 0.818 | 0.817 | 0.795 | 0.780 | 0.821 | **0.840** |
| **Avg. Rank** | 4 | 3.7 | 4.3 | 3.6 | 3.2 | **2.2** | 3.4 | 3.6 | 4.3 | 4.7 | 2.9 | **2.1** |

classifiers and 25 imbalanced data sets. In these Tables, the second-to-last row shows the total average results of all data sets ($Avg.Val$) for each resampling method, and the last row shows the average rank ($Avg.Rank$) in the computation of the Friedman test [12] for each resampling method over all data sets. For the $Avg.Rank$, the lowest at $Avg.Rank$ indicated the best ranking of all representative methods.

Table V presents the AUC results of all representative methods, which are trained by AdaBoost-C4.5 and AdaBoost-NB. For the results from the AUC of AdaBoost C4.5 classifier, our proposed COTU approach obtained the highest total average value ($Avg.Val$) at $90.9\%$ for all test data sets, with the highest average rank ($Avg.Rank$) value $2.2$. For AdaBoost-NB classifier, our proposed approach obtained the highest results on $8$ out of $25$ data sets and the highest total average value ($Avg.Val$) was $88.5\%$ overall test data sets, resulting in the best average rank ($Avg.Rank$) with a value of $2.6$. But, SMOTE-TL and Center-NN take the second place in both classifiers, while SBC is the last.

Similarly, Table VI shows the G-mean results for all representative methods. For AdaBoost-C4.5 classifier, COTU approach obtained the highest results on $13$ out of $25$ data sets and the highest total average value ($Avg.Val$) overall test data sets at $84.9\%$, resulting in the best average rank ($Avg.Rank$) with a value of $2$. For AdaBoost-NB, COTU approach achieved the highest results on $12$ out of $25$ data sets and the highest total average value ($Avg.Val$) at $82.3\%$ overall test data sets, with the best average rank ($Avg.Rank$) value $2.1$.

In Table VII, considering the BA results for AdaBoost-C4.5 and AdaBoost-NB classifiers, our proposed COTU approach achieved the best overall average values ($Avg.Val$) at $85.9\%$ and $84\%$ respectively, and obtained the best average ranking ($Avg.Rank$) among all tested models. While, SMOTE-TL performed better on the other resampling methods, especially AUC and BA results.

According to the details results from Tables V to VII, COTU approach has achieved better results and rankings than other resampling techniques, in terms of AUC, G-mean and BA results. Besides, the results from Tables V to VII indicated that both true positive and true negatives had been relatively increased by COTU approach.

### *B. Probabilistic Estimation by Ensemble-Based Learning Classifiers*

In the second experimental analysis, the probabilistic prediction, Balanced Brier score ($BS^{Bal}$) is used as a measure of uncertainty, which calculates the probability estimates of observed data for all representative methods. In the probabilistic predictions of all resampling methods, the smaller $BS^{Bal}$ result is more accurate than the larger $BS^{Bal}$ result because it measures the distance between the probabilities of the predicted class value and the actual class value.

TABLE VIII
$BS^{Bal}$ RESULTS FOR 25 DATA SETS AND SIX DATA PRE-PROCESSING TECHNIQUES USING ADABOOST-C4.5 AND ADABOOST-NB CLASSIFIERS

| Data Sets | AdaBoost-C4.5 | | | | | | AdaBoost-NB | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | SMOTE | Center - NN | SBC | BDSK | SMOTE - TL | COTU | SMOTE | Center - NN | SBC | BDSK | SMOTE - TL | COTU |
| glass1 | 0.362 | 0.393 | 0.669 | 0.432 | 0.413 | **0.346** | 0.421 | **0.42** | 0.625 | 0.436 | 0.449 | 0.428 |
| vehicle1 | 0.512 | 0.481 | 0.493 | 0.458 | 0.506 | **0.44** | **0.392** | 0.396 | 0.445 | 0.428 | 0.408 | 0.404 |
| vehicle3 | 0.52 | 0.467 | 0.462 | 0.53 | 0.455 | **0.445** | **0.389** | 0.39 | 0.424 | 0.418 | 0.391 | 0.392 |
| vehicle0 | **0.073** | 0.093 | 0.467 | 0.077 | 0.081 | 0.118 | 0.168 | 0.205 | 0.443 | 0.196 | 0.173 | **0.161** |
| new-thyroid1 | 0.093 | 0.08 | 0.096 | 0.117 | **0.051** | **0.051** | 0.037 | 0.028 | 0.063 | 0.057 | **0.02** | 0.032 |
| newthyroid2 | 0.116 | 0.097 | 0.145 | 0.089 | 0.097 | **0.087** | 0.047 | 0.041 | 0.06 | **0.012** | 0.05 | 0.027 |
| segment0 | 0.011 | 0.017 | 0.175 | **0.01** | 0.017 | 0.018 | 0.032 | 0.09 | 0.242 | **0.023** | 0.034 | 0.061 |
| glass6 | 0.173 | 0.151 | 0.2 | 0.188 | 0.175 | **0.14** | 0.258 | 0.254 | 0.121 | 0.261 | 0.179 | **0.113** |
| yeast3 | 0.229 | 0.187 | 0.192 | 0.21 | **0.146** | 0.159 | 0.157 | 0.164 | 0.209 | 0.214 | **0.149** | 0.159 |
| ecoli-034_Vs_5 | **0.122** | 0.168 | 0.243 | 0.135 | 0.193 | 0.238 | **0.194** | 0.26 | 0.212 | 0.222 | 0.222 | 0.243 |
| yeast-2_Vs_4 | 0.233 | 0.168 | 0.163 | 0.219 | **0.156** | 0.163 | 0.211 | 0.195 | 0.237 | 0.265 | 0.234 | **0.162** |
| yeast-02579_Vs_368 | 0.2 | 0.231 | 0.251 | 0.221 | **0.186** | 0.206 | 0.188 | 0.233 | 0.287 | 0.343 | 0.207 | **0.183** |
| ecoli-0267_Vs_35 | 0.295 | **0.266** | 0.376 | 0.275 | 0.299 | 0.272 | **0.231** | 0.31 | 0.399 | 0.476 | 0.236 | 0.36 |
| ecoli-067_Vs_5 | 0.344 | 0.335 | 0.247 | **0.154** | 0.282 | 0.233 | **0.321** | 0.352 | 0.326 | **0.321** | 0.324 | 0.35 |
| led7digit-02456789_Vs_1 | **0.182** | 0.202 | 0.253 | 0.221 | 0.719 | 0.198 | 0.249 | **0.205** | 0.285 | 0.276 | 0.217 | 0.212 |
| glass-0146_Vs_2 | 0.632 | 0.658 | 0.603 | 0.689 | **0.301** | 0.46 | 0.519 | 0.552 | 0.641 | 0.589 | 0.551 | **0.479** |
| glass2 | 0.481 | 0.802 | 0.92 | 0.639 | 0.585 | **0.316** | 0.488 | 0.563 | 0.608 | 0.542 | 0.507 | **0.413** |
| ecoli4 | 0.266 | 0.163 | 0.301 | **0.068** | 0.216 | 0.184 | 0.18 | 0.111 | 0.089 | 0.175 | 0.215 | **0.08** |
| glass-016_Vs_5 | 0.211 | 0.103 | 0.114 | 0.119 | 0.217 | **0.076** | **0.158** | 0.161 | 0.247 | 0.475 | 0.178 | 0.186 |
| yeast-2_Vs_8 | 0.488 | 0.523 | 0.533 | 0.412 | **0.365** | 0.435 | **0.442** | 0.528 | 0.609 | 0.487 | 0.475 | 0.478 |
| yeast4 | 0.562 | 0.402 | **0.341** | 0.529 | 0.456 | 0.366 | 0.318 | 0.328 | 0.345 | 0.452 | 0.338 | **0.313** |
| winequality-white-9_Vs_4 | 0.425 | 0.372 | 0.594 | 0.419 | 0.423 | **0.351** | 0.617 | **0.378** | 0.93 | 0.606 | 0.613 | 0.385 |
| ecoli-0137_Vs_26 | 0.515 | 0.429 | 0.46 | 0.511 | **0.321** | 0.414 | 0.314 | 0.347 | 0.426 | 0.514 | 0.323 | **0.27** |
| yeast6 | 0.383 | 0.282 | 0.287 | 0.391 | 0.384 | **0.279** | **0.226** | 0.274 | 0.344 | 0.344 | 0.242 | 0.228 |
| winequality-white-39_Vs_5 | 0.854 | 0.772 | 0.751 | 0.929 | 0.793 | **0.658** | **0.432** | 0.486 | 0.613 | 0.745 | 0.495 | 0.491 |
| **Avg. Val** | 0.331 | 0.314 | 0.373 | 0.322 | 0.313 | **0.266** | 0.280 | 0.291 | 0.369 | 0.355 | 0.289 | **0.264** |
| **Avg. Rank** | 4 | 3.3 | 4.6 | 3.8 | 3.3 | **2.1** | **2.3** | 3.3 | 5.1 | 4.6 | 3.3 | 2.4 |

Table VIII shows the Brier score results for all representative methods obtained by AdaBoost-C4.5 and AdaBoost-NB. For AdaBoost-C.45 classifier, our proposed COTU approach is more successful on 11 out of 25 data sets than the other five methods of resampling, and the lowest total average value ($Avg.Val$) is 26.6%, resulting in the best average rank ($Avg.Rank$) with a value of 2.1. As for AdaBoost-NB classifier, the lowest average $BS^{Bal}$ result ($Avg.Val$) at 26.4% has been obtained from COTU approach.

The results of this experiment indicated that COTU approach was successful not only in the class label prediction results (i.e. AUC, G-mean and BA), but also in the estimation of the probability (i.e. $BS^{Bal}$).

### *C. Statistical Analysis by Ensemble-Based Learning Classifiers*

In the third experimental analysis, the statistical significance tests of COTU approach are presented using Wilcoxon's Signed Rank Test and the detail results are shown in Tables IX, X, XI and XII. In these Tables, the $Win–Tie-Loss$ statistics by summarizing the results, all positive-sum ranks ($W^+$), all negative-sum ranks ($W^-$) and $P_{value}$ of Wilcoxon's Signed Rank Tests between the pairs of COTU and comparative resampling methods based on 25 imbalanced data sets are presented. The performance of two resampling methods is significantly different if $P_{value}$ is less than 0.05.

In Tables IX, X, XI and XII, all positive-sum ranks ($W^+$) are higher than the negative-sum ranks ($W^-$) in every pair. In both two classifiers, COTU approach performed significantly better than the other resampling methods, in terms of AUC, G-mean, BA and $BS^{Bal}$ results, but COTU approach does not differ in AdaBoost-C4.5's SMOTE-TL method, especially AUC, BA and $BS^{Bal}$ results.

### *D. Determination of Best Sampling Method by Ensemble-Based Learning Classifiers*

In the fourth experimental analysis, we suggested the best resampling method for each learning classifier by implementing TOPSIS strategy in the MCDM methodology [38], [40]. To rank sampling methods for each classifier, the performance of six resampling methods of all data sets is entered in TOPSIS method. For each classifier, we used the six alternatives (sampling methods) with 25 criteria (data sets) because the effectiveness of each resampling method depends on the performance of all data sets.

Table XIII reports the relative closeness to the ideal solution and the ranking of all resampling methods in terms of AdaBoost-C4.5 and AdaBoost-NB. In these tables, the higher ideal solution value is more accurate than the lower ideal solution, in terms of AUC, G-mean and BA results. But, the lower ideal solution of $BS^{Bal}$ is more accurate than the higher ideal solution value because the smaller $BS^{Bal}$ result is more accurate than the larger $BS^{Bal}$ result.

TABLE IX
WILCOXON'S SIGNED RANK TEST OF AUC RESULTS OF ADABOOST-C4.5 AND ADABOOST-NB CLASSIFIERS

| Methods | AdaBoost-C4.5 | | | | AdaBoost-NB | | | |
|---|---|---|---|---|---|---|---|---|
| | $W^+$ | $W^-$ | $Win-Tie-Loss$ | $P_{value}$ | $W^+$ | $W^-$ | $Win-Tie-Loss$ | $P_{value}$ |
| COTU Vs SMOTE | 219 | 81 | 15-1-9 | **0.0486** | 230 | 70 | 16-1-8 | **0.0223** |
| COTU Vs Center-NN | 273 | 27 | 19-1-5 | **0.0004** | 240.5 | 84.5 | 17-0-8 | **0.0358** |
| COTU Vs SBC | 325 | 0 | 25-0-0 | **0.0000** | 261 | 39 | 18-1-6 | **0.0015** |
| COTU Vs BDSK | 240.5 | 84.5 | 17-0-8 | **0.0358** | 258 | 42 | 18-1-6 | **0.0020** |
| COTU Vs SMOTE-TL | 199 | 77 | 17-2-6 | 0.0634 | 206 | 70 | 14-2-9 | **0.0385** |

TABLE X
WILCOXON'S SIGNED RANK TEST OF G-MEAN RESULTS OF ADABOOST-C4.5 AND ADABOOST-NB CLASSIFIERS

| Methods | AdaBoost-C4.5 | | | | AdaBoost-NB | | | |
|---|---|---|---|---|---|---|---|---|
| | $W^+$ | $W^-$ | $Win-Tie-Loss$ | $P_{value}$ | $W^+$ | $W^-$ | $Win-Tie-Loss$ | $P_{value}$ |
| COTU Vs SMOTE | 293 | 32 | 20-0-5 | **0.0004** | 258.5 | 66.5 | 18-0-7 | **0.0098** |
| COTU Vs Center-NN | 261.5 | 38.5 | 20-1-4 | **0.0014** | 250.5 | 49.5 | 19-1-5 | **0.0041** |
| COTU Vs SBC | 260 | 16 | 21-2-2 | **0.0002** | 302.50 | 22.50 | 21-0-4 | **0.0002** |
| COTU Vs BDSK | 266 | 59 | 20-0-5 | **0.0054** | 311 | 14 | 21-0-4 | **0.0001** |
| COTU Vs SMOTE-TL | 188.5 | 64.5 | 13-3-7 | **0.0441** | 258.50 | 66.50 | 18-0-7 | **0.0098** |

TABLE XI
WILCOXON'S SIGNED RANK TEST OF BA RESULTS OF ADABOOST-C4.5 AND ADABOOST-NB CLASSIFIERS

| Methods | AdaBoost-C4.5 | | | | AdaBoost-NB | | | |
|---|---|---|---|---|---|---|---|---|
| | $W^+$ | $W^-$ | $Win-Tie-Loss$ | $P_{value}$ | $W^+$ | $W^-$ | $Win-Tie-Loss$ | $P_{value}$ |
| COTU Vs SMOTE | 287 | 38 | 20-0-5 | **0.0008** | 250.5 | 74.5 | 18-0-7 | **0.0178** |
| COTU Vs Center-NN | 242 | 34 | 19-2-4 | **0.0016** | 249 | 51 | 18-1-6 | **0.0047** |
| COTU Vs SBC | 312.5 | 12.5 | 21-0-4 | **0.0001** | 302 | 23 | 21-0-4 | **0.0002** |
| COTU Vs BDSK | 227 | 73 | 18-1-6 | **0.0278** | 310 | 15 | 21-0-4 | **0.0001** |
| COTU Vs SMOTE-TL | 190.5 | 85.5 | 15-2-8 | 0.1103 | 239 | 86 | 18-0-7 | **0.0394** |

TABLE XII
WILCOXON'S SIGNED RANK TEST OF $BS^{Bal}$ RESULTS OF ADABOOST-C4.5 AND ADABOOST-NB CLASSIFIERS

| Methods | AdaBoost-C4.5 | | | | AdaBoost-NB | | | |
|---|---|---|---|---|---|---|---|---|
| | $W^+$ | $W^-$ | $Win-Tie-Loss$ | $P_{value}$ | $W^+$ | $W^-$ | $Win-Tie-Loss$ | $P_{value}$ |
| COTU Vs SMOTE | 289.5 | 35.5 | 20-0-5 | **0.0006** | 191 | 134 | 13-0-12 | 0.4430 |
| COTU Vs Center-NN | 277 | 48 | 20-0-6 | **0.0021** | 253 | 72 | 16-0-9 | **0.0149** |
| COTU Vs SBC | 295 | 5 | 23-1-1 | **0.0000** | 317.5 | 7.5 | 23-0-2 | **0.0000** |
| COTU Vs BDSK | 252.5 | 72.5 | 19-0-6 | **0.0154** | 300 | 25 | 21-0-4 | **0.0002** |
| COTU Vs SMOTE-TL | 210 | 90 | 15-1-9 | 0.0865 | 233 | 92 | 16-0-9 | 0.0578 |

TABLE XIII
RELATIVE CLOSENESS TO IDEAL SOLUTION AND FINAL RANKING FOR 25 DATA SETS AND SIX DATA PRE-PROCESSING TECHNIQUES USING ADABOOST-C4.5 AND ADABOOST-NB CLASSIFIERS

| Methods | | Relative Closeness to Ideal Solution | | | | | | Final Ranking | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | SMOTE | Center - NN | SBC | BDSK | SMOTE - TL | COTU | SMOTE | Center - NN | SBC | BDSK | SMOTE - TL | COTU |
| $AUC$ | AdaBoost-C4.5 | 0.7 | 0.403 | 0.133 | 0.707 | 0.755 | **0.77** | 4 | 5 | 6 | 3 | 2 | **1** |
| | AdaBoost-NB | 0.381 | 0.675 | 0.35 | 0.516 | 0.447 | **0.813** | 5 | 2 | 6 | 3 | 4 | **1** |
| $G-mean$ | AdaBoost-C4.5 | 0.453 | 0.644 | 0.528 | 0.355 | 0.507 | **0.867** | 5 | 2 | 3 | 6 | 4 | **1** |
| | AdaBoost-NB | 0.587 | 0.811 | 0.428 | 0.273 | 0.577 | **0.91** | 3 | 2 | 5 | 6 | 4 | **1** |
| $BA$ | AdaBoost-C4.5 | 0.57 | 0.45 | 0.352 | 0.511 | 0.547 | **0.784** | 2 | 5 | 6 | 4 | 3 | **1** |
| | AdaBoost-NB | 0.624 | 0.656 | 0.42 | 0.349 | 0.631 | **0.849** | 4 | 2 | 5 | 6 | 3 | **1** |
| $BS^{Bal}$ | AdaBoost-C4.5 | 0.358 | 0.289 | 0.68 | 0.292 | 0.387 | **0.204** | 4 | 2 | 6 | 3 | 5 | **1** |
| | AdaBoost-NB | 0.315 | 0.323 | 0.679 | 0.481 | 0.319 | **0.184** | 2 | 4 | 6 | 5 | 3 | **1** |

In Table XIII, based on the relative closeness to ideal solution and the final ranking of all learning classifiers and resampling methods, COTU approach ranked first because it obtained rank 1 for each classifier. Especially, COTU approach combined with AdaBoost-C4.5 and AdaBoost-NB classifiers will significantly improve performance compared to the five well-known resampling methods, while SBC and BDSK are last-rank.

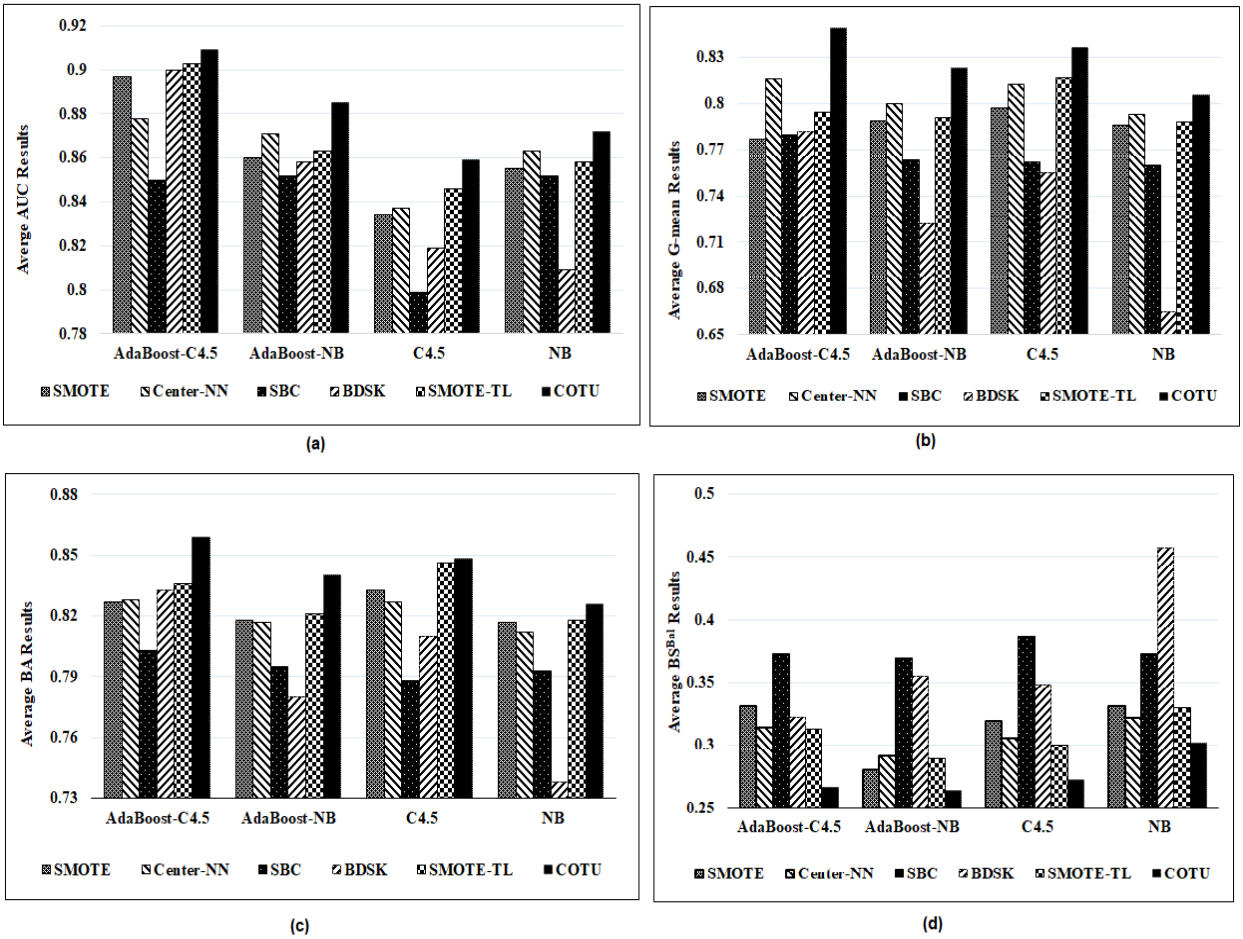


Fig. 3. Comparison of six pre-processing techniques for the twenty-five imbalanced data sets trained by the classifiers; AdaBoost-C4.5, AdaBoost-NB, C4.5 and NB (a) Average AUC Results (b) Average G-mean Results (c) Average BA (Balanced Accuracy) Results (d) Average $BS^{Bal}$ (Brier score) Results

TABLE XIV
AVERAGE VALUE AND AVERAGE RANK OF AUC, G-MEAN, BA AND $BS^{Bal}$ RESULTS FOR FOUR CLASSIFIERS

| Methods | | Avg. Val | | | | | | Avg. Rank | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | SMOTE | Center - NN | SBC | BDSK | SMOTE - TL | COTU | SMOTE | Center - NN | SBC | BDSK | SMOTE - TL | COTU |
| $AUC$ | AdaBoost-C4.5 | 0.897 | 0.878 | 0.85 | 0.9 | 0.903 | **0.909** | 3 | 3.9 | 4.9 | 3.6 | 3.4 | **2.2** |
| | AdaBoost-NB | 0.86 | 0.871 | 0.852 | 0.858 | 0.863 | **0.885** | 3.2 | 3.6 | 4.1 | 4.1 | 3.3 | **2.6** |
| | C4.5 | 0.834 | 0.837 | 0.799 | 0.819 | 0.846 | **0.859** | 3.4 | 3.3 | 4.7 | 3.8 | 3.3 | **2.4** |
| | NB | 0.855 | 0.863 | 0.852 | 0.809 | 0.858 | **0.872** | 3.7 | 3.1 | 3.3 | 4.8 | 3.2 | **2.8** |
| $G-mean$ | AdaBoost-C4.5 | 0.777 | 0.816 | 0.78 | 0.782 | 0.794 | **0.849** | 4.2 | 3.5 | 4.4 | 3.7 | 3.2 | **2** |
| | AdaBoost-NB | 0.789 | 0.8 | 0.763 | 0.722 | 0.791 | **0.823** | 3.2 | 3.5 | 4.3 | 4.9 | 3 | **2.1** |
| | C4.5 | 0.797 | 0.813 | 0.762 | 0.755 | 0.817 | **0.836** | 3.5 | 3 | 4.8 | 4.3 | 2.9 | **2.5** |
| | NB | 0.786 | 0.793 | 0.76 | 0.665 | 0.788 | **0.806** | **2.6** | 3.4 | 4.3 | 5.1 | 2.7 | 2.8 |
| $BA$ | AdaBoost-C4.5 | 0.827 | 0.828 | 0.803 | 0.833 | 0.836 | **0.859** | 4 | 3.7 | 4.3 | 3.6 | 3.2 | **2.2** |
| | AdaBoost-NB | 0.818 | 0.817 | 0.795 | 0.78 | 0.821 | **0.84** | 3.4 | 3.6 | 4.3 | 4.7 | 2.9 | **2.1** |
| | C4.5 | 0.833 | 0.827 | 0.788 | 0.81 | 0.846 | **0.848** | 3.3 | 3.3 | 5 | 4.1 | 2.7 | **2.6** |
| | NB | 0.817 | 0.812 | 0.793 | 0.738 | 0.818 | **0.826** | **2.6** | 3.4 | 4.4 | 5.1 | 2.7 | 2.8 |
| $BS^{Bal}$ | AdaBoost-C4.5 | 0.331 | 0.314 | 0.373 | 0.322 | 0.313 | **0.266** | 4 | 3.3 | 4.6 | 3.8 | 3.3 | **2.1** |
| | AdaBoost-NB | 0.28 | 0.291 | 0.369 | 0.355 | 0.289 | **0.264** | **2.3** | 3.3 | 5.1 | 4.6 | 3.3 | 2.4 |
| | C4.5 | 0.319 | 0.306 | 0.386 | 0.348 | 0.3 | **0.272** | 3.4 | 3 | 5 | 4 | 3.1 | **2.5** |
| | NB | 0.331 | 0.322 | 0.373 | 0.457 | 0.33 | **0.302** | 2.8 | 3.4 | 4.5 | 5 | 3.1 | **2.2** |

### *E. Overall Results by Four Learning Classifiers*

In the last experimental analysis, the average performance scores of all representative methods implemented by AdaBoost-C4.5 and AdaBoost-NB, C4.5 and NB classifiers are presented. Fig. 3 shows the graphical representation of average value and average ranks for all data sets, using six resampling methods and four learning classifiers. In addition, to clear the representation of graphs, the notations used in

TABLE XV
WILCOXON'S SIGNED RANK TEST RESULTS FOR FOUR CLASSIFIERS

| Methods | | COTU Vs SMOTE | COTU Vs Center-NN | COTU Vs SBC | COTU Vs BDSK | COTU Vs SMOTE-TL |
|---|---|---|---|---|---|---|
| $AUC$ | AdaBoost-C4.5 | **0.0486** | **0.0004** | **0.0000** | **0.0358** | 0.0634 |
| | AdaBoost-NB | **0.0223** | **0.0358** | **0.0015** | **0.0020** | **0.0385** |
| | C4.5 | **0.0133** | **0.0164** | **0.0001** | **0.0214** | 0.1657 |
| | NB | **0.0394** | 0.7422 | 0.2528 | **0.0005** | **0.0296** |
| $G-mean$ | AdaBoost-C4.5 | **0.0004** | **0.0014** | **0.0002** | **0.0054** | **0.0441** |
| | AdaBoost-NB | **0.0098** | **0.0041** | **0.0002** | **0.0001** | **0.0098** |
| | C4.5 | **0.0185** | 0.0632 | **0.0001** | **0.0094** | 0.5011 |
| | NB | 0.3259 | 0.2056 | **0.0003** | **0.0000** | 0.1887 |
| $BA$ | AdaBoost-C4.5 | **0.0008** | **0.0016** | **0.0001** | **0.0278** | 0.1103 |
| | AdaBoost-NB | **0.0178** | **0.0047** | **0.0002** | **0.0001** | **0.0394** |
| | C4.5 | **0.0450** | **0.0485** | **0.0001** | **0.0255** | 0.9893 |
| | NB | 0.4841 | 0.2105 | **0.0002** | **0.0000** | 0.3833 |
| $BS^{Bal}$ | AdaBoost-C4.5 | **0.0006** | **0.0021** | **0.0000** | **0.0154** | 0.0865 |
| | AdaBoost-NB | 0.4430 | **0.0149** | **0.0000** | **0.0002** | 0.0578 |
| | C4.5 | **0.0128** | 0.0926 | **0.0000** | **0.0119** | 0.1228 |
| | NB | 0.1388 | **0.0494** | **0.0001** | **0.0000** | **0.0197** |

TABLE XVI
RELATIVE CLOSENESS TO IDEAL SOLUTION AND FINAL RANKING FOR FOUR CLASSIFIERS

| Methods | | Relative Closeness to Ideal Solution | | | | | | Final Ranking | | | | | |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| | | SMOTE | Center - NN | SBC | BDSK | SMOTE - TL | COTU | SMOTE | Center - NN | SBC | BDSK | SMOTE - TL | COTU |
| $AUC$ | AdaBoost-C4.5 | 0.7 | 0.403 | 0.133 | 0.707 | 0.755 | **0.77** | 4 | 5 | 6 | 3 | 2 | **1** |
| | AdaBoost-NB | 0.381 | 0.675 | 0.35 | 0.516 | 0.447 | **0.813** | 5 | 2 | 6 | 3 | 4 | **1** |
| | C4.5 | 0.582 | 0.497 | 0.296 | 0.392 | 0.694 | **0.821** | 3 | 4 | 6 | 5 | 2 | **1** |
| | NB | 0.648 | 0.725 | 0.57 | 0.243 | 0.662 | **0.842** | 4 | 2 | 5 | 6 | 3 | **1** |
| $G-mean$ | AdaBoost-C4.5 | 0.453 | 0.644 | 0.528 | 0.355 | 0.507 | **0.867** | 5 | 2 | 3 | 6 | 4 | **1** |
| | AdaBoost-NB | 0.587 | 0.811 | 0.428 | 0.273 | 0.577 | **0.91** | 3 | 2 | 5 | 6 | 4 | **1** |
| | C4.5 | 0.595 | 0.61 | 0.412 | 0.276 | 0.657 | **0.868** | 4 | 3 | 5 | 6 | 2 | **1** |
| | NB | 0.648 | 0.845 | 0.5 | 0.099 | 0.647 | **0.905** | 3 | 2 | 5 | 6 | 4 | **1** |
| $BA$ | AdaBoost-C4.5 | 0.57 | 0.45 | 0.352 | 0.511 | 0.547 | **0.784** | 2 | 5 | 6 | 4 | 3 | **1** |
| | AdaBoost-NB | 0.624 | 0.656 | 0.42 | 0.349 | 0.631 | **0.849** | 4 | 2 | 5 | 6 | 3 | **1** |
| | C4.5 | 0.633 | 0.463 | 0.284 | 0.371 | 0.717 | **0.791** | 3 | 4 | 6 | 5 | 2 | **1** |
| | NB | 0.762 | 0.735 | 0.563 | 0.196 | 0.77 | **0.853** | 3 | 4 | 5 | 6 | 2 | **1** |
| $BS^{Bal}$ | AdaBoost-C4.5 | 0.358 | 0.289 | 0.68 | 0.292 | 0.387 | **0.204** | 4 | 2 | 6 | 3 | 5 | **1** |
| | AdaBoost-NB | 0.315 | 0.323 | 0.679 | 0.481 | 0.319 | **0.184** | 2 | 4 | 6 | 5 | 3 | **1** |
| | C4.5 | 0.368 | 0.277 | 0.73 | 0.373 | 0.349 | **0.226** | 4 | 2 | 6 | 5 | 3 | **1** |
| | NB | 0.262 | 0.304 | 0.428 | 0.666 | 0.233 | **0.172** | 3 | 4 | 5 | 6 | 2 | **1** |

Fig. 3 is represented in Table XIV.

Based on the average results of all experiments in Table XIV and Fig. 3: (a) to (d), the results obtained by COTU approach are not found to be significantly lower than any other resampling methods. In particular, COTU approach combined with AdaBoost-C4.5 remains at the top of four learning models, and it's $BS^{Bal}$ result is lower than the other three learning classifiers. But, the combination of NB classifier and BDSK remains at the last position of all learning classifiers, and it's $BS^{Bal}$ result is larger than the other three learning classifiers.

The $P_{value}$ of Wilcoxon's Signed Rank Test shown in Table XV, COTU approach can significantly increase among all representative methods in AdaBoost-NB classifier, but it does not differ with SMOTE-TL in AdaBoost-C4.5 and C4.5 classifiers.

Based on the final ranking of four learning classifiers for all resampling methods shown in Table XVI, COTU approach achieved the first-rank in all learning models. COTU approach, combined with AdaBoost-C4.5 classifiers will significantly improve performance compared with the five well-known resampling methods, while AdaBoost-NB classifier takes the second place.

In order to the above results of all experiments and statistical tests, we can conclude that the efficiency of the learning classifier for imbalanced data is predominated by the class distribution. Moreover, COTU's results are not found to be significantly lower than any other resampling methods since COTU approach can overcome the three characteristics of class distribution problems between classes.

## VI. CONCLUSION AND FUTURE WORK

This paper had proposed an effective resampling approach (COTU) to address the problems of skewed distribution in

the imbalanced data sets. Firstly, the proposed over-sampling step based on k-means clustering was used to handle the small disjuncts problem in the minority class. As a second step, Tomek Link-based under-sampling method was also integrated among cluster-based sampling steps to remove the overlapping majority samples. The cluster-based under-sampling step was also used as the final step of COTU approach for solving the problem of small sample size in the training set. After executing the proposed approach, the refining training data set was fed into the classification phase to predict unknown input variables.

In the experimental analysis, we evaluated the performance of each resampling method by using the four different learners. And, the performance of our proposed approach was validated with the three-class label predictive metrics (such as AUC, G-mean and BA), as well as the expected probabilistic cost estimates ($BS^{Bal}$). The performance of data sets varied according to the methods of resampling and learning classifiers. Our results show that COTU approach outperformed the five state-of-the-art resampling methods. Moreover, the performance of COTU approach was significantly improved among the representative methods according to the $P_{value}$ of Wilcoxon's Signed Rank Test. And, the best sampling method was suggested by implementing TOPSIS method.

In future work, we plan to expand COTU approach so that the multi-class imbalanced data can be classified. Moreover, an optimal value of clusters ($K$) will be specified for the problems of small disjuncts and small sample size, as the number of clusters ($K$) influences the efficiency of classification.

## REFERENCES


[1] A. Agrawal, H. L. Viktor and E. Paquet, "SCUT: Multi-class imbalanced data classification using SMOTE and cluster-based under-sampling," *In 2015 7th International Joint Conference on Knowledge Discovery, Knowledge Engineering and Knowledge Management (IC3K), IEEE* vol. 1, pp226-234, 2015.

[2] M. Y. Arafat, S. Hoque, S. Xu and D. M. Farid, "Machine Learning for Mining Imbalanced Data," *IAENG International Journal of Computer Science*, vol. 46, no. 2, pp332-348, 2019.

[3] G. E. Batista, R. C. Prati and M. C. Monard, "A study of the behavior of several methods for balancing machine learning training data," *ACM SIGKDD explorations newsletter*, vol. 6, no. 1, pp20-29, 2004.

[4] M. Bekkar, H. K. Djemaa, and T. A. Alitouche, "Evaluation measures for models assessment over imbalanced data sets," *J Inf Eng Appl*, vol. 3, no. 10, pp27-38, 2013.

[5] G. W. Brier, "Verification of forecasts expressed in terms of probability," *Monthly weather review*, vol. 78, no. 1, pp1-3, 1950.

[6] K. H. Brodersen, C. S. Ong, K. E. Stephan, and J. M. Buhmann, "The balanced accuracy and its posterior distribution," *In 2010 20th International Conference on Pattern Recognition, IEEE*, pp3121-3124, 2010.

[7] C. Bunkhumpornpat, K. Sinapiromsaran and C. Lursinsap, "Safe-level-smote: Safe-level-Synthetic minority over-sampling technique for handling the class imbalanced problem," *In Pacific-Asia conference on knowledge discovery and data mining*, pp475-482, 2009.

[8] N. V. Chawla, K. W. Bowyer, L. O. Hall and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," *Journal of artificial intelligence research*, vol. 16, pp321-357, 2002.

[9] L. Chen, B. Fang, Z. Shang and Y. Tang, "Tackling class overlap and imbalance problems in software defect prediction," *Software Quality Journal*, vol. 26, no. 1, pp97-125, 2018.

[10] T. Choeikiwong and P. Vateekul, "Two Stage Model to Detect and Rank Software Defects on Imbalanced and Scarcity Data Sets," *IAENG International Journal of Computer Science*, vol. 43, no. 3, pp344-355, 2016.

[11] D. Devi and B. Purkayastha, "Redundancy-driven modified Tomek-link based under-sampling: a solution to class imbalance," *Pattern Recognition Letters*, vol. 93, pp3-12, 2017.

[12] J. Demšar, "Statistical comparisons of classifiers over multiple data sets," *Journal of Machine learning research*, vol. 7, pp1-30, 2006.

[13] G. Douzas, F. Bacao and F. Last, "Improving imbalanced learning through a heuristic over-sampling method based on K Means and SMOTE," *Information Sciences*, vol. 465, pp1-20, 2018.

[14] T. Fawcett, "An introduction to ROC analysis," *Pattern recognition letters*, vol. 27, no. 8, pp861-874, 2006.

[15] S. Fotouhi, S. Asadi and M. W. Kattan, "A comprehensive data level analysis for cancer diagnosis on imbalanced data," *Journal of biomedical informatics*, vol. 90, 2019.

[16] Y. Freund and R. E. Schapire, "A decision-theoretic generalization of on-line learning and an application to boosting," *In European conference on computational learning theory* pp23-37, 1995.

[17] M. Galar, A. Fernandez, E. Barrenechea, H. Bustince, and F. Herrera, "A review on ensembles for the class imbalanced problem: bagging-, boosting-, and hybrid-based approaches," *IEEE Transactions on Systems, Man, and Cybernetics, Part C (Applications and Reviews)*, vol. 42, no. 4, pp463-484, 2011.

[18] H. Han, W. Y. Wang and B. H. Mao, "Borderline-SMOTE: a new over-sampling method in imbalanced data sets learning," *In International conference on intelligent computing, Lecture Notes in Computer Science*, vol. 3644, pp878-887, 2005.

[19] G. Haixiang, L. Yijing, J. Shang, G. Mingyun, H. Yuanyue and G. Bing, "Learning from class-imbalanced data: Review of methods and applications," *Expert Systems with Applications*, vol. 73, pp220-239, 2017.

[20] M. Hall, E. Frank, G. Holmes, B. Pfahringer, P. Reutemann and I. H. Witten, "The WEKA data mining software: an update," *ACM SIGKDD explorations newsletter*, vol. 11, no. 1, pp10-18, 2009.

[21] J. Han, M. Kamber and J. Pei, "Data mining concepts and techniques third edition," *Morgan Kaufmann*, 2011.

[22] P. Hart, "The condensed nearest neighbor rule (Corresp.)," *IEEE transactions on information theory*, vol. 14, no. 3, pp515-516, 1968.

[23] H. He and E. A. Garcia, "Learning from imbalanced data," ÎEEE Transactions on Knowledge & Data Engineering, no. 21, vol. 9, pp1263-1284, 2009.

[24] T. Jo and N. Japkowicz, "Class imbalances versus small disjuncts," *ACM Sigkdd Explorations Newsletter*, vol. 6, no. 1, pp40-49, 2004.

[25] Q. Kang, X. Chen, S. Li and M. Zhou, "A noise-filtered under-sampling scheme for imbalanced classification," *IEEE transactions on cybernetics*, vol. 47, no. 12, pp4263-4274, 2016.

[26] R. Kohavi, "A study of cross-validation and bootstrap for accuracy estimation and model selection," *In Ijcai* vol. 14, no. 2, pp1137-1145, 1995.

[27] M. Koziarski and M. Woźniak, "CCR: A combined cleaning and resampling algorithm for imbalanced data classification," *International Journal of Applied Mathematics and Computer Science*, vol. 27, no. 4, pp727-736, 2017.

[28] M. Kubat, and S. Matwin, "Addressing the curse of imbalanced training sets: one-sided selection," *In Icml*, vol. 97, pp179-186, 1997.

[29] J. Laurikkala, "Improving identification of difficult small classes by balancing class distribution," *In Conference on Artificial Intelligence in Medicine in Europe*, pp63-66, 2001.

[30] W. C. Lin, C. F. Tsai, Y. H. Hu and J. S. Jhang, "Clustering-based under-sampling in class-imbalanced data," *Information Sciences*, vol. 409, pp17-26, 2017.

[31] X. Y. Liu, J. Wu, and Z. H. Zhou, "Exploratory under-sampling for class-imbalance learning," *IEEE Transactions on Systems, Man, and Cybernetics, Part B (Cybernetics)*, vol. 39, no. 2, pp539-550, 2008.

[32] M. M. Nwe and K. T. Lynn, "KNN-Based Overlapping Samples Filter Approach for Classification of Imbalanced Data," *In International Conference on Software Engineering Research, Management and Applications, Springer, Cham*, vol. 845, pp55-73, 2019.

[33] N. Ofek, L. Rokach, R. Stern and A. Shabtai, "Fast-CBUS: A fast clustering-based under-sampling method for addressing the class imbalanced problem," *Neurocomputing*, vol. 243, pp88-102, 2017.

[34] A. Onan, "Consensus Clustering-Based under-sampling Approach to Imbalanced Learning," *Scientific Programming*, vol. 2019, pp1-14, 2019.

[35] W. A. Rivera, "Noise reduction a priori synthetic over-sampling for class imbalanced datasets," *Information Sciences*, vol. 408, pp146-161, 2017.

[36] J. A. Sáez, J. Luengo, J. Stefanowski and F. Herrera, "SMOTE–IPF: Addressing the noisy and borderline examples problem in imbalanced classification by a resampling method with filtering," *Information Sciences*, vol. 291, pp184-203, 2015.

[37] S. L. Salzberg, "C4.5: Programs for machine learning by j. ross quinlan. morgan kaufmann publishers, inc., 1993," *Machine Learning*, pp235-240, 1994.

[38] R. Singh, H. Kumar and R. K. Singla, "TOPSIS based multi-criteria decision making of feature selection techniques for network traffic dataset," *International Journal of Engineering and Technology*, vol. 5, no. 6, pp4598-4604, 2014.
[39] J. Song, X. Huang, S. Qin and Q. Song, "A bi-directional sampling based on K Means method for imbalance text classification," *IEEE/ACIS 15th International Conference on Computer and Information Science (ICIS)*, pp1-5, 2016.
[40] Y. Song and Y. Peng, "A MCDM-Based Evaluation Approach for Imbalanced Classification Methods in Financial Risk Prediction," *IEEE Access*, vol. 7, pp84897-84906, 2019.
[41] R. A. Sowah, M. A. Agebure, G. A. Mills, K. M. Koumadi and S. Y. Fiawoo,"New cluster under-sampling technique for class imbalanced learning," *International Journal of Machine Learning and Computing*, vol. 6, no. 3, pp205, 2016.
[42] Y. Sun and F. Liu, "SMOTE-NCL: A resampling method with filter for network intrusion detection," *2nd IEEE International Conference on Computer and Communications (ICCC)*, pp1157-1161, 2016.
[43] I. Tomek, "Two modifications of CNN," *IEEE Trans, Systems Man and Cybernetics*, vol. 6, pp769-772, 1976.
[44] D. R. Velez, B. C. White, A. A. Motsinger, W. S. Bush, M. D. Ritchie, S. M. Williams, and, J. H. Moore, "A balanced accuracy function for epistasis modeling in imbalanced datasets using multifactor dimensionality reduction," *Genetic Epidemiology: the Official Publication of the International Genetic Epidemiology Society*, vol. 31, no. 4, pp306-315, 2007.
[45] P. Vuttipittayamongkol and E. Elyan, "Neighbourhood-based undersampling approach for handling imbalanced and overlapped data," *Information Sciences*, vol. 509, pp47-70, 2020.
[46] C. Wang and Y. Yang, "Nearest Neighbor with Double Neighborhoods Algorithm for Imbalanced Classification," *IAENG International Journal of Applied Mathematics* , vol. 50, no. 1, pp147-159, 2020.
[47] B. C. Wallace and I. J. Dahabreh, "Improving class probability estimates for imbalanced data," *Knowledge and information systems*, vol. 41, no. 1, pp33-52, 2014.
[48] F. Wilcoxon, "Some rapid approximate statistical procedures," *Annals of the New York Academy of Sciences*, vol. 52, no. 1, pp808-814, 1950.
[49] D. L. Wilson, "Asymptotic properties of nearest neighbor rules using edited data," *IEEE Transactions on Systems, Man, and Cybernetics*, no. 3, pp408-421, 1972.
[50] S. J. Yen and Y. S. Lee, "Cluster-based approaches for imbalanced data distributions," *Expert Systems with Applications*, vol. 36, no. 3, pp5718-5727, 2009.
[51] L. Zhou, "Performance of corporate bankruptcy prediction models on imbalanced dataset: The effect of sampling methods," *Knowledge-Based Systems*, vol. 41, pp16-25, 2013.