International Journal of Computer Information Systems and Industrial Management Applications.
ISSN 2150-7988 Volume 14 (2022) pp. 151-161
© MIR Labs, www.mirlabs.net/ijcisim/index.html

Received: 1 July, 2021; Accepted: 3 February, 2022; Publish: 24 April, 2022

OptDCE: An Optimal and Diverse Classifier Ensemble for Imbalanced Datasets

Uma R. Godase[1], Darshan V. Medhane[2]

[1] Department of Computer Engineering, International Institute of Information Technology,
Pune, Maharashtra, India
urgodase@gmail.com

[2] Department of Computer Engineering, MVPS's KBT College of Engineering,
Nashik, Maharashtra, India
darshan.medhane@gmail.com

***Abstract*: Machine learning has evolved dramatically in recent years and plays a very important role to ease the day-to-day activities. Classification is one of the major tasks in machine learning. It is concerned with the categorization of the data in various applications such as software fault detection, credit scoring systems and medical applications. Many of these applications suffer from the problem of Imbalanced data classification wherein one class consists of a large number of samples while samples representing another class are very less in number. The skewed nature of data results in the imprecise classification of the data which may be very harmful in some disciplines like medical applications. To highlight the class imbalance issue, this work presents the impact of the increased degree of class imbalance on the classification performance of various datasets. Moreover, we present the classification approach that integrates the data level technique with a diverse classifier ensemble (CE). The experimental results show significant improvements in the classification performance of imbalanced datasets.**

Keywords: Imbalanced Data, Re-sampling, Classifier ensemble, Diversity, Machine Learning**.**

I. Introduction

The classification of data, concerned with the categorization of the data is one of the most commonly used tasks in machine learning. The supervised learning approach of classification begins by training the model with the available training data comprising the various attributes as well as class labels [1]. This results in building the predictive model that is further tested on the unseen data to check its accuracy. The model having fewer misclassification errors is treated as a good predictive model. Lots of work is carried out to improve the accuracy of the classification and significant improvements in the prediction accuracy are seen. However, some of the research works did not pay attention to the attributes of the data that has been used to train the model. One such attribute that plays a significant role in the classification accuracy is the imbalanced nature of the training data. This work deals with the disparity between the two classes for binary classification.

The remainder of this paper is organized as follows. Initially, the change in the classification performance against degree of imbalance is highlighted. Further, the next section presents the summary of the associated problems with class imbalance, the possible solutions and their impact on the performance. A novel approach for classification is introduced to deal with these problems. Finally, the experimental results on various imbalanced datasets show the significantly improved performance.

A. Class Imbalance Problem

Most of the classification systems encompass the training data with skewed nature. Few examples to mention are churn prediction systems, credit scoring systems, or medical applications. In these datasets, the majority class has a high number of samples while the minority class consists of a very small number of samples. During the training phase of the classifier using such an imbalanced dataset, the majority class plays a prominent role [2]. As a result, the predictive model is likely to be inclined towards the majority class. That is, the unseen instances of minority class may be classified as members of the majority class. Figure 1 demonstrates such a scenario wherein the decision-making process of the predictive model is biased.

The class imbalance issue discussed in this section may lead to misclassification of most of the minority class samples. Practically, the cost of misclassification of minority class is much greater than that of the majority class [3]. Especially in the medical field, it may probably result in serious consequences such as harm to the life of the patient that is misclassified with a negative diagnosis. Therefore, imbalanced data learning is grabbing the attention of the researchers. This work is targeted towards handling the same issue.

B. Classifier performance against the degree of imbalance

This section presents how the performance of classification algorithms is influenced due to the class imbalance issue. The experiments are designed by varying the degree of imbalance of sample datasets to examine the extent to which the classifier performance gets affected. The experimentation was carried out on a Car dataset that is publicly available in the

University of California at Irvine (UCI) repository.

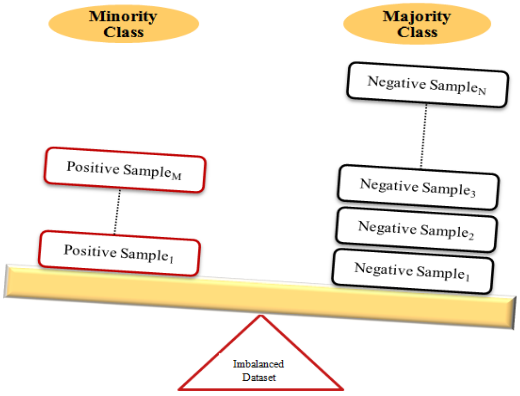


Figure 1. Biased Decision towards Majority Class

Total 13 subsets with different degrees of imbalance were generated by using this dataset. The measure used to indicate the degree of imbalance is called "Imbalance Ratio "(IR) which can be defined as the ratio of the number of instances of the majority class to that of the minority class [4]. Out of these 13 subsets used for the experimentation, one was the actual dataset having an original IR while the remaining were having different IR. To vary their level of imbalance, few instances of the minority class were randomly deleted. A total of four classification algorithms were used for the experiments out of which two were the individual classification algorithms namely Decision Tree and Support Vector Machine (SVM) while the remaining two were classifier ensemble (CE) techniques namely Bootstrap aggregating (Bagging) and Adaptive Boosting (AdaBoost). The CE is constructed by combining different individual classifiers that are diversely intending to improve the performance [5]. The purpose behind using them was to examine the impact of the increased Imbalance degree on the individual classification algorithm against the impact on the CE performance. For this, the amount of degraded performance of the individual classifier was compared with that of the CE.

Table 1 presents the Area Under ROC curve (AUC) values obtained for 13 subsets created using the Car dataset. The results were obtained using 10-fold cross-validation. The Imbalance Degree of these subsets is in the range of 26 to 333.

Two CEs used in this study belong to two categories based on how they are formed. That is, the Bagging ensemble is formed by combining diverse base classifiers that are trained in parallel. On the contrary, AdaBoost is formed by combining the different classifiers trained one after the other wherein every next classifier emphasizes the instances misclassified by the previous classifier [6]. This parallel vs sequential construction of the ensemble results in less creation time for the Bagging ensemble whereas relatively more construction time for the AdaBoost ensemble approach. Figure 2 represents the AUC values of 13 subsets of the Car dataset for four classification algorithms namely J48, SVM, Bagging, and AdaBoost.

The careful observation of the graph shown in Figure 2 derives some very interesting conclusions. The line for each classifier signifies that increasing IR values of the dataset result in declining values of the AUC measure.

Sr. No.	IR	J48	SVM	Bagging	AdaBoost
1	26	0.992	0.92	0.994	0.998
2	28	0.993	0.913	0.995	0.999
3	30	0.994	0.842	0.995	0.999
4	33	0.911	0.887	0.996	0.998
5	37	0.472	0.785	0.99	0.998
6	42	0.499	0.809	0.978	0.998
7	48	0.464	0.797	0.985	0.985
8	55	0.499	0.915	0.979	0.995
9	67	0.449	0.919	0.989	0.997
10	83	0.499	0.923	0.913	0.993
11	111	0.416	0.799	0.436	0.987
12	166	0.499	0.699	0.5	0.992
13	333	0.25	0.5	0.26	0.995

Table 1. AUC at different Imbalance Ratios for Car dataset

Though all the classifiers show degraded performance due to the increased imbalance degree, the amount of degradation seen is significantly different. Individual classifiers J48 and SVM show a sudden drop in their AUC values when the IR crosses 30. Obtained AUC values for J48 are dropped from 0.992 to 0.25 whereas AUC in the range of 0.92 to 0.5 is observed for SVM. On the other hand, the performance degradation seen in the case of Bagging and AdaBoost is relatively very less. The AdaBoost gives 0.999 maximum AUC value while the minimum value of AUC is 0.985. Therefore, the line representing AdaBoost in the above graph is almost horizontal. Initially, AUC values for Bagging classifiers show a gradual decrease but as IR value exceeds 110, a significant drop in the obtained AUC values is observed. Therefore, the almost horizontal line is slanted at the end.

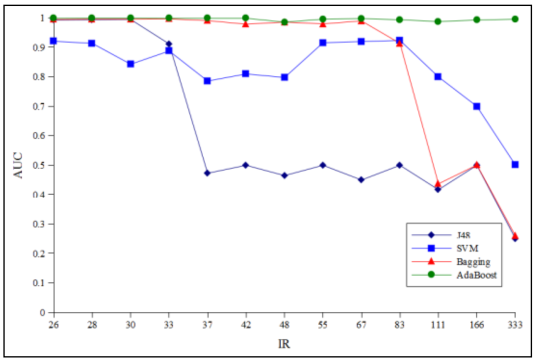


Figure 2. Impact of Imbalance Ratio Evaluated on Car Dataset

The motive of the above experimentation was to inspect how the individual classification algorithms and CE techniques respond to the changes in the level of imbalance of the imbalanced datasets. Findings from the analysis of the obtained experimental results are as follows:

1. For all the datasets on which experiments are conducted, the graph of IR versus AUC shows the significant degradation in AUC values due to IR. As shown in Figure 3, as the degree of imbalance measured by the IR increases, the performance of the classification measured in AUC keeps on decreasing. Thus, the imbalance present in the training datasets

plays a very crucial role in classification performance. Therefore, special attention must be given to this issue.

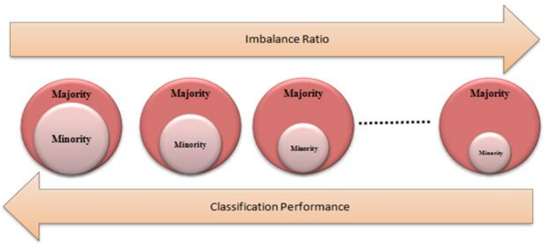


Figure 3. Classification Performance against Degree of Imbalance

2. The comparison of the AUC values for individual classifiers against that of CE clearly shows that CE techniques give significantly better performance despite the imbalanced distribution of the datasets. To be specific, as the degree of imbalance between the classes increases, the AUC values for individual classification algorithms are drastically declined. That is, the sudden drop in AUC values is observed. On the other hand, the amount of degradation observed for the CE techniques is relatively less. Also, it has been observed that instead of a sudden drop, AUC values are decayed gradually. Thus, the CE proves to be a preferable technique to handle the higher imbalance degree.
3. Two CEs selected for the experimentation are different from each other in their method of formation. The purpose was to verify their behavior in the presence of class imbalance. Resultant AUC values clearly show the better performance of the AdaBoost CE than the Bagging ensemble. However, their sequential versus parallel construction approach results in very high training times for the AdaBoost ensemble. Therefore, considering the tradeoff between the classifier performance and time required to train the model, the proposed work presents the Bagging based ensemble.

II. Related Work

A thorough review of the state-of-the-art research work in the area of the imbalanced datasets was carried out to find out the challenges that are faced by the existing work and still need to be addressed. This section gives an overview of the existing work carried out towards the class imbalance issue.

The clustering algorithm, Semantic Driven Subtractive Clustering Method (SDSCM) for customer churn management is developed by authors [7]. The use of Axiomatic Fuzzy Sets (AFS) algebra and structure allows expressing complex concepts with the help of many simple concepts. A parallel SDSCM is implemented on the Hadoop MapReduce framework to test the proposed method for China telecom big data. This has shown significant speed improvements.

Adnan Amin et al. [8] presented an intelligent rule-based decision-making technique for customer churn prediction. The rough set theory (RST) is used as a basis for mining important decision rules. RST is applied in combination with different rule-generation approaches namely the Exhaustive Algorithm (EA), Genetic Algorithm (GA) and Covering Algorithm (CA) algorithm. The performance evaluation in terms of precision, recall, the rate of misclassification, lift, coverage, accuracy, and F-measure shows better performance of an approach that combines RST with GA. Also, RST and EA combination was found inefficient because the produced decision rules were not useful.

Ammar A. Q. Ahmed et al. [9] dealt with a huge telecom dataset with imbalanced nature. The firefly algorithm is enhanced by replacing the comparison part with Simulated Annealing to reduce the computation time. Simulated Annealing gets the optimum solution by identifying the firefly with maximum intensity. The application of Firefly and Hybrid firefly algorithm on orange dataset shows that the accuracy of Firefly gets slightly increased from 86.36% to 86.38% due to hybridization. The authors suggest further enhancements of the proposed approach so that False Positive (FP) rates get decreased. Yanmin Sun et al. [6] enhanced the AdaBoost algorithm by including the concept of cost sensitiveness to tackle the class imbalance exhibited by many real-life applications. Three variants called AdaC1, AdaC2, and AdaC3 are proposed wherein the weight parameter is modified taking into account the costs associated with each class. This results in a final predictive model with minimum training error. The results show that AdaC2 and AdaC3 get higher recall than precision which may not be true for AdaC1. These two algorithms are sensitive to the values assigned to cost.

Xin Xia et al. [10] constructed a feature-level CE called imbalanced Multi Label K-Nearest Neighbor (Im-ML.KNN) by enhancing Multi Label K-Nearest Neighbor (ML.KNN) to predict which bug report fields will be reassigned and refined. Different types of features like Meta, textual, and mixed are used to train the base classifiers and generate diverse learning models. It is found that Im-ML.KNN, when compared with other reference techniques shows F-measure improvement by 119.69%, 9.11%, and 161.08% respectively. Also, it improves the F-measure of its base classifiers namely meta-classifier, text classifier, and mixed classifier by 8.91%, 164.31%, and 9.11%, respectively. Varying the number of neighbors does not have a significant impact on the performance of the algorithm. The approach needs to be tested on a variety of bug reports from various projects.

Abdulla Amin Aburomman et al. [11] applied a CE technique to the intrusion detection system. Particle swarm optimization (PSO) and meta-optimized PSO approaches are used to construct the ensembles. Evaluation of knowledge discovery and data mining 1999(KDD99) dataset in terms of accuracy shows the increase in accuracy of a base expert by 0.756%. But the accuracy may not reflect the performance of the minority class.

Zhen Liu et al. [12] focused on the generalization capacity of the classification algorithm for network classification in a technique known as SMOTEAdaNL. They made combined use of the re-sampling technique with the boosting-based ensemble. The weight of each sample is modified based on two terms namely error rate and penalty term. A penalty term is concerned with the diversity of base classifiers of the ensemble. Thus, the samples that are misclassified and have low disagreement levels are given more weight. The performance of SMOTEAdaNL is compared with the performance of weighted re-sampling (WRS), flow size modernization (FSM), and the combination of ensemble learning and cost-sensitive learning ECS. The significant improvements are seen in the results. But the proposed approach targets only wired traffic and does not consider wireless traffic.

Lei Bao et al. [13] developed a Boosted Near-miss Under-sampling on SVM ensembles (BNU-SVM) technique in which each iteration selects the nearest miss examples of a minority class. Additionally, it focuses on computational complexity issue raised due to high dimensional data. To handle it, a kernel-distance precomputation technique is also proposed. G. Vinodhini et al. proposed a hybrid approach based on the data level and CE approach [14] for sentiment mining. The proposed M-Bagging approach uses bootstrapping with replacement and Synthetic Minority Oversampling Technique (SMOTE) as a re-sampling method. Generated bags are used to generate diverse base classifiers. Q-statistic is used to measure the diversity between members of the ensemble. The presented SVM-based ensemble has shown significant improvements in the performance of minority class as well. But it may not deal with a high degree of imbalance.

Nazim Bushara1 et al. [15] proposed a novel CE based weather forecasting model for rainfall prediction. It includes vote meta classifier that combines three base classifiers. The proposed CE increases the accuracy of prediction as well as results in greater confidence in the results.

A. Handling Imbalanced Data: Problems, Solutions, and Effects

To summarize the associated problems of the imbalanced dataset, Table 2 describes the associated problems with imbalanced data distribution, their possible solutions, and the resulting consequences.

As illustrated in Table 2, one approach to handle the class imbalance involves the conversion of the imbalanced dataset into a relatively balanced form. To achieve this, re-sampling techniques are designed that incorporates modification of the original imbalanced class distribution to convert the uneven distribution of data into even distribution. To increase the size of the minority class, oversampling of the data is done while the proportion of the majority class can be reduced with the help of under-sampling. The purpose of oversampling is to expand the minority class either by duplicating the existing instances or adding new artificial instances. Significant attention has been given by many researchers to overcome their limitations. However, relatively less interest is seen in another category. That is, under-sampling which ignores some of the instances of the majority class is mostly done by the random selection of the instances that are to be ignored. Moreover, there has been little focus to overcome the limitations of random under-sampling. Therefore, the fact that the necessary data of the majority class gets removed is ignored and may result in performance degradation. The proposed classification scheme is designed with the perspective of dealing with this issue and focuses on the necessary data of the majority class.

The CE approaches are designed to enhance the classification performance by consulting multiple individual classification algorithms. However, identical experts may not lead to improved performance [16]. As a result, the overhead is increased but significant performance gains are not achieved [17]. Therefore, there is a need to ensure the diversity of the combined base classifiers by making use of diversity measures. Another issue with the CEs is unnecessarily larger ensembles that increase the computational overhead. This necessitates the construction of an optimal CE that achieves maximum performance benefit as well as maintains the smaller size.

Sr. No.	Approach	Associated Problem	Possible Solution	Effects
1	Data Level	Uneven distribution of data - The bigger size of the majority class	Conversion of the balanced form using under-sampling of the majority class	Probability of removal of important data
2	Data Level	Uneven distribution of data - The smaller size of the minority class	Conversion of the balanced form using oversampling of the minority class	Over-fitting
3	Classifier Ensemble	Non-diverse base learners - No enhancements in performance	Make use of diversity measure	Enhanced accuracy
4	Classifier Ensemble	Larger ensemble size - Increased computational overhead	Ensemble with smaller size but good performance	Reduced storage requirement
5	Algorithm Level	Algorithm unsuitable for imbalanced data	Customized modification in specific algorithm	Effectiveness depends on the choice of the learning algorithm and problem domain
6	Cost-sensitive Learning	Same misclassification cost to the majority and minority class	Higher misclassification cost to the minority class	Improper cost assignment may lead to degradation of performance

Table 2. Summary of Associated Problems with Imbalanced Data, Solutions and Effects

III. Proposed Methodology

The key objective of the proposed work is to overcome the few challenges identified in the existing work to enhance the performance of the classifier designed for datasets with skewed nature. To accomplish this, a data level technique named Borderline Under-sampling (BLUS) [18] is integrated with a novel CE approach that is designed considering performance improvement as well as optimum size of the ensemble. The proposed work is carried out in two phases as shown in Figure 4.

In the first phase of the proposed work, the concern is to reduce the degree of the imbalance between the two classes which involves the reduction of the IR either by increasing the size of the minority class or decreasing the size of the majority class. The process is called Re-sampling of the dataset. The researchers in the domain had recommended that combined use of under-sampling and oversampling approaches proves to be more effective than making use of any one of them [16].

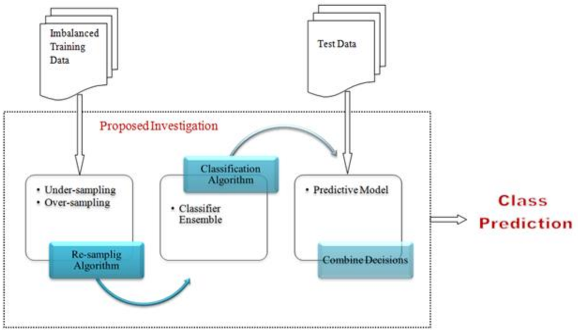


Figure 4. Phases of the proposed methodology

Therefore, the proposed work initially applies oversampling to the minority class to artificially introduce the new instances. This involves the use of one of the most commonly used approaches in the existing work known as SMOTE [19]. Further, under-sampling of the majority class is done with the help of BLUS which involves the removal of a few instances from the class without removing any necessary data of the class. The peculiarity of the BLUS approach is its focus on important data of the class so that such instances are identified and guarded against random deletion in the under-sampling phase. That is, the instances present near the decision boundary play a critical role in defining the decision boundary between the two classes. Therefore, the instances in this region are considered more important and should be retained in the dataset used to train the predictive model.

The second phase involves the formation of the CE targeted to provide the maximum performance with the optimal size of the ensemble. The output of the first phase is a dataset that is balanced to some extent relative to the original imbalanced dataset. This balanced dataset is used in the second phase to train the predictive model. As discussed in the previous section, the CE is the predictive model that is formed by combining the predictions of multiple classifiers. The theory behind consulting multiple experts is exploiting their strengths in decision making and giving the final prediction based on the predictions offered by the majority of them. That is, it is less likely that most of them agree on the incorrect decision which in turn will improve the accuracy of the prediction. Thus, taking the majority votes from the different individual classification algorithms will improve the classification performance. However, if the classifiers that are combined are similar, the probability of improving the performance is less. To be more precise, the similar nature of the individual classifiers leads to similar decisions by them which in turn does not improve the performance. Therefore, there is a need to ensure the design of a diverse set of individual classifiers. Moreover, the number of base classifiers to be combined should not be very high because after a certain limit adding the new base classifiers may not improve the performance. To be more precise, it will result in the additional overhead of combining the results from too many experts without significant performance gains. Considering all those factors, the proposed CE aims to combine a diverse set of base classifiers keeping the size of the ensemble optimal.

A. Phase 1: Pre-processing of Imbalanced Data Using Data Level Technique (PID-DLT)

The pre-processing phase is executed to modify the proportion of the instances belonging to the majority and minority class. The intention behind this is to reduce the degree of imbalance between the two classes so that the cardinality of the two classes becomes approximately equal. In the proposed work, the imbalance between the two classes is reduced by modifying the cardinality of the two classes. For this, both under-sampling of the majority class, as well as oversampling of the minority class, is done. Moreover, before re-sampling the imbalanced data, the noisy instances from the input data are removed so that they don't affect the performance of the classifier. Thus, pre-processing involves two important processes:

1. Identification and removal of the noisy instances by comparing with nearby instances.
2. To balance the datasets using the under-sampling and oversampling technique.

The Algorithm titled PID-DLT takes an imbalanced dataset as an input, applies the two phases of pre-processing, and gives a balanced dataset as an output.

Algorithm 1 Pre-processing of Imbalanced Data using Data Level Technique (PID-DLT)

```
Input: Imbalanced training dataset $T_{IMBAL}$
Output: Re-sampled dataset $T_{BAL}$
Initialize
        $B_{IMBAL} \leftarrow$ Majority-Class ($T_{IMBAL}$)
        $S_{IMBAL} \leftarrow$ Minority-Class ($T_{IMBAL}$)
Procedure Pre - process
Start
    1: Read the imbalanced dataset $T_{IMBAL}$
    2: Identify the noisy instances in a set $B_{NOISY}$ as
        $B_{NOISY} \leftarrow$ NIIR ($B_{IMBAL}$)
    3: Remove $B_{NOISY}$ from $B_{IMBAL}$ as
        $B'_{IMBAL} \leftarrow B_{IMBAL} - B_{NOISY}$
    4: Re-sample the $T_{IMBAL}$
        Over-sample $S_{IMBAL}$
        $S_{OVER} \leftarrow$ SOS ($S_{IMBAL}$)
        Under-sample $B'_{IMBAL}$
        $B_{UNDER} \leftarrow$ BLUS ($B'_{IMBAL}$)
    5: $T_{BAL} \leftarrow S_{OVER} \cup B_{UNDER}$
    6: Return $T_{BAL}$
Stop
```

A brief description of the various steps in algorithm PID-DLT is as follows:

1. The algorithm receives the training dataset in the imbalanced form as an input and executes a set of pre-processing operations to generate the balanced dataset as an output.

2. The presence of noise in the given dataset is taken into account as it may harm the performance of the classification algorithm. The procedure initiates by identifying such noisy instances from the training dataset. This is done in the procedure NIIR and the resultant noisy instances are returned into set B_{NOISY}.

3. The members of set B_{NOISY} should not be considered further to train the learning model. Hence, the dataset B_{NOISY} needs to be removed from the original training dataset and the remaining dataset should be taken into account for further

processing. The set B'$_{\text{IMBAL}}$ comprise of the majority class of the training data after taking out the noisy data and can be represented as

$$B'_{\text{IMBAL}} = \{h \mid h \in B_{\text{IMBAL}} \wedge h \notin B_{\text{NOISY}}\} \quad (1)$$

4. Now, the re-sampling of the remaining training dataset is done. The set S$_{\text{IMBAL}}$ is expanded by invoking Synthetic Minority oversampling (SOS) procedure which returns the result in S$_{\text{OVER}}$. BLUS procedure is invoked which reduces the size of B$_{\text{IMBAL}}$ and returns the under-sampled result in B$_{\text{UNDER}}$.
5. The sets B$_{\text{UNDER}}$ and S$_{\text{OVER}}$ that are generated in the above steps are now combined to form the balanced dataset T$_{\text{BAL}}$.
6. The generated T$_{\text{BAL}}$ is returned.
The diagrammatic representation of the conversion of imbalanced training data set T$_{\text{IMBAL}}$ to T$_{\text{BAL}}$ using algorithm PID-DLT is as shown in Figure 5.

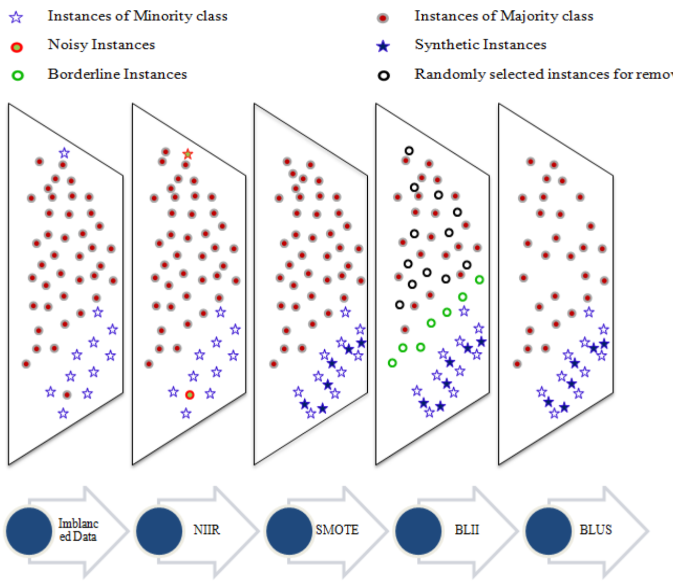

☆ Instances of Minority class
● Instances of Majority class
● Noisy Instances
★ Synthetic Instances
○ Borderline Instances
○ Randomly selected instances for remo
Imblanced Data
NIIR
SMOTE
BLII
BLUS

Figure 5. Illustration of the Re-sampling phase

B. *Phase 2: Classification using Classifier Ensemble*

Initially, the mini ensembles are constructed which will work as base classifiers for the second phase. The basic idea is concerned with taking advantage of the diversity between the classifiers but without unnecessarily creating larger ensembles. To be more precise, increasing the number of base classifiers may not result in the increased accuracy of the ensemble. Because of this, the proposed approach takes the set of mini ensembles ME and selects some of them as base classifiers for the second level ensemble. The base classifiers that are selected have the highest diversity value than any other combination of base classifiers. Further, predictions of the selected mini ensembles are combined to get the final prediction. Algorithm OptDCE explains the process for selecting the diverse and optimum number of mini ensembles that should be combined to give the final prediction of class labels.

Algorithm 2 Optimal and Diverse Classifier Ensemble (OptDCE)

```
Input: Set of Mini-Ensembles ME
Output: Final prediction of the class label
Procedure OptimalEnsemble
Initialize
        S ← Size of a subset of Mini-Ensembles
        P(ME)= { ϕ }
        Numsub ← $\binom{L}{S}$
Start
    1: Read the given set ME.
    2: for k = 1 : $\binom{L}{2}$, i; j ∈ {1,2…..L } do
    $Disagreement_{ME_iME_j} = \frac{A^{01}_{i,j} + A^{10}_{i,j}}{\sum_{xy\in\{11,10,01,00\}} A^{xy}_{i,j}}$
      end for
    3: Generate power set of ME
        for p = 1 : $2^L$ do
        $Subset_p$ = { $ME_i$ | $ME_i$ ∈ ME ∧ i ∈{ 1,2…...L } }
        end for
        P(ME) = { $Subset_p$ | $Subset_p$ ⊂ ME }
        Select the subsets of smaller size
            Final_Subset = { $Subset_p$ | $Subset_p$ ⊆ P(ME)
            ∧ |$Subset_p$| = S }
    4: for $Subset_z$ ∈ Final_Subset, z = 1 : Numsub do
        $AvgDis_z = \frac{2}{S(S-1)} \sum_{i,j\in\{1,2...L\} \wedge ME_iME_j \in Subset_z}^{\binom{L}{2}} Disagree_{ME_iME_j}$
        end for
    5: Find the subset that represents the set of mini
      ensembles with the highest diversity.
     ∃ $Subset_{max}$(∀ q < Numsub, $AvgDis_q$ < $AvgDis_{max}$)
    6: Construct final ensemble as
          FE = arg max [$ME_i$]
              such that $ME_i$ ∈ $Subset_{max}$
Stop
```

Detailed Steps:

1. Read the set of mini ensembles generated by using different training sets and different base classifiers.
2. The next step is to calculate the diversity between the classifiers. The numerous diversity measures discussed in the existing literature belong to two categories namely pairwise and non-pairwise diversity measures. We have used a pairwise measure known as disagreement [20] that can be defined as the fraction of the instances that are predicted differently by the pair of classifiers. To be specific, let us assume that MEi and MEj are the two mini ensembles for which the disagreement factor is to be calculated. Then the class prediction of each instance by both of them is compared and the instance is added to one of the four values of the contingency table. The contingency table shown in Table 3 gives an abstract view of the outputs of a pair of mini ensembles MEi and MEj. Each value in the table represents the number of instances that are correctly or incorrectly classified by the pair. $A^{xy}_{i,j}$ is the count of the instances for which classifier MEi predicts the label x while classifier MEj predicts as y. For example, $A^{10}_{i,j}$ represents the number of instances that are correctly classified by MEi and incorrectly classified by MEj.

	Correct (1)	**Incorrect (0)**

Correct (1)	$A_{i,j}^{11}$	$A_{i,j}^{10}$
Incorrect (0)	$A_{i,j}^{01}$	$A_{i,j}^{00}$

Table 3. Contingency Table for a Pair of Mini-Ensembles

To be more precise, member of the contingency matrix for the classifiers MEi and MEj can be represented as

$$A_{i,j}^{xy} = \sum_{k=0}^{n} p(T_k^{xy}) \quad (2)$$

The function p in the above equation can be formulated as

$P(Q^{xy})$ = 1 if Q is classified as x by MEi and y by MEj
= 0 Otherwise

where

x, y: Class labels
xy ∈ {11,10,01,00}

The values of the contingency table for any pair of classifiers are used to derive the disagreement between these two classifiers. The disagreement between the two classifiers MEi and MEj is computed as follows:

$$Disagreement_{ME_iME_j} = \frac{A_{i,j}^{01} + A_{i,j}^{10}}{\sum_{xy\in\{11,10,01,00\}} A_{i,j}^{xy}} \quad (3)$$

The above process is repeated for each possible combination of two classifiers from the set of mini ensembles. That is, disagreement for each MEi and MEj where values of i and j are between 1 and L is calculated. With L classifiers the number of possible pairs will be $\binom{L}{2}$ and hence the pairwise disagreement is calculated for all of them.

3. The next step is finding the different possible subsets of a set ME. i.e. Subset1, Subset2 … Subset Numsub. This power set contains 2L possible subsets out of which the process selects the subsets of size S. From a set of L classifiers, the various combinations that can be generated are

$$Numsub = \binom{L}{S} \quad (4)$$

The selected subsets are the final subsets that are checked for the highest diversity.

4. The subsequent steps deal with diversity analysis using disagreement as a diversity measure. Once the different possible subsets of the set ME are ready, the average disagreement for each possible combination is calculated as

$$AvgDis_z = \frac{2}{S(S-1)} \sum_{i,j\in\{1,2\ldots L\} \wedge ME_iME_j \in Subset_z}^{\binom{L}{2}} Disagree_{ME_iME_j} \quad (5)$$

5. The highest value of AvgDis$_z$ represents the subset Subsetmax that contains the mini ensembles that are more diverse than any other possible combination. It comprises all the mini ensembles that are diverse and optimum for getting enhanced classifier performance. Therefore, members of Subset$_{max}$ are selected to participate in the final decision-making process.

6. The final ensemble model that combines the predictions of selected mini ensembles is defined as

$$FE = argmax\left[sign \sum_{t=1}^{iter} l(\,Best_i, D_i(t), x)\right] \quad (6)$$

The proposed OptDCE approach is an integrated framework that makes use of the proposed CE with a novel re-sampling technique. Figure 6 depicts how the re-sampling and classification algorithms work together to handle the skewed datasets that are usually seen in many applications nowadays.

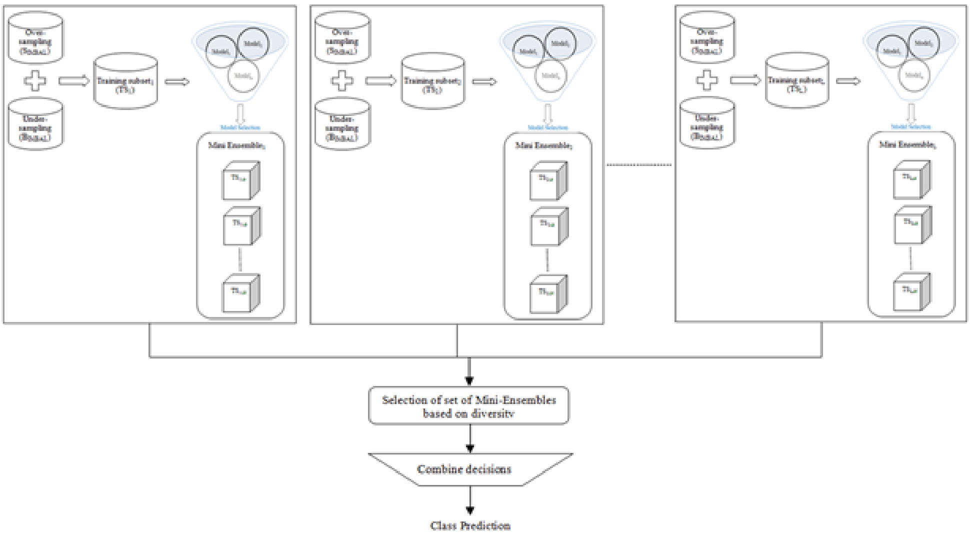


Figure 6. Proposed Classifier Ensemble Using Borderline Under-sampling

As shown in Figure 6, the research work involves two imbalance handling methods namely data level approach and CE technique. The data level technique targets at decreasing the class imbalance by modifying the original data distribution. A hybrid approach of re-sampling is employed to the training data so that the number of instances of two classes becomes approximately equal. A well-known technique SMOTE is exploited to over-sample the minority class while the novel methodology BLUS has been applied to under-sample the majority class. The outputs of over-sampling and under-sampling are combined to form the training subsets. The process is repeated L times so that L training subsets are constructed that are diverse and can be used to build the diverse classification models.

The generated training subsets are used to construct a pool of base classifiers which are then evaluated to choose the most appropriate classifier for the training subset under consideration. Further, the selected best classifier is treated as a base classifier of the mini ensemble formed at the first layer of ensembles. A set of mini ensembles created in this manner works as an input to the second layer of the CE.

The number of mini ensembles generated in the previous phase need not be included in the final ensemble construction. To be specific, some of these mini ensembles may be identical and hence may not contribute to improving the performance. Therefore, the diverse set of mini ensembles is selected in the second layer. The selection is done based on the diversity between the set of mini ensembles that is denoted by the average of the pairwise diversity measure known as disagreement. Finally, the predictions of selected mini ensembles are combined to generate a final prediction. Thus, a final class label is predicted which is a correct prediction for the given instance.

III. Result Analysis and Discussions

The next set of experiments was designed to evaluate the proposed system that integrates the proposed SOS - BLUS technique with a novel CE. To accomplish this, the input training data is initially pre-processed with SOS – BLUS method which is then used to construct a novel CE. The final results of the proposed OptDCE technique are validated against various state-of-the-art classification approaches that are specially aimed at handling the imbalanced data. The series of experiments were conducted to assess and compare the proposed work concerning different types of existing imbalance handling methods. The detailed experimental results are discussed in the following subsections.

A. Experimental Setup

The experimentation was carried out using Weka (Waikato Environment for Knowledge Analysis) environment version 3.6 with its default parameters. Weka is an open-source toolkit that offers a library of various machine learning as well as pre-processing algorithms. The proposed methodology has been implemented in Java and the experiments were carried out using 5-fold cross-validation.B. Experimental Datasets

For experimentation, we have chosen various datasets that are publicly available in the UCI repository and Knowledge Extraction based on Evolutionary Learning (KEEL) repository. The chosen datasets comprise the data with a skewed nature. The selected datasets are diverse in a way that they have a varying degree of imbalance and the type of data involved is different as well.

C. Comparison with Bagging-based Techniques

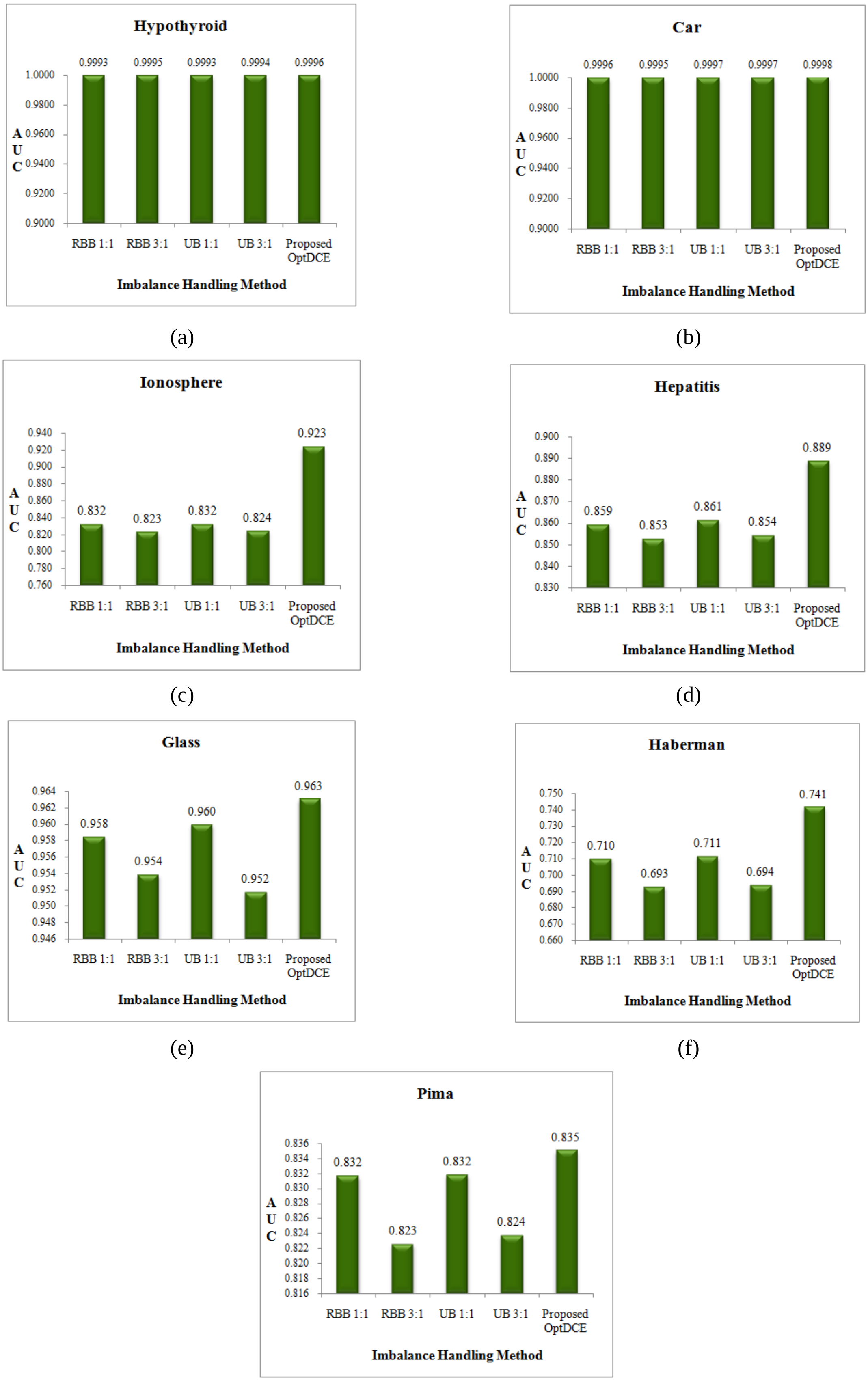
This section offers the comparison of the proposed OptDCE with existing imbalance handling methods that make combined use of the data level and the CE techniques. Furthermore, the ensembles generated in these approaches are bagging based in which the members of the CE are constructed in parallel and the individual predictions of members are combined to generate the final prediction. The first set of experiments was performed to compare the proposed work with four imbalance handling methods that are based on the data level and bagging-based CE technique. Seven imbalanced datasets with different IRs are used for the experimentation. The four reference techniques selected for the comparison are as follows [21]:

- Roughly Balanced Bagging [RBB 1:1]
- Roughly Balanced Bagging [RBB 3:1]
- Under Bagging [UB 1:1]
- Under Bagging [UB 3:1]

Figure 7 illustrates the AUC results of the proposed methodology compared against the four reference techniques listed above. The results for various datasets are presented in each sub-graph. Table 4 represents these results. An analysis of the graphs indicates that the proposed OptDCE outperforms the other imbalance handling methods and achieves greater AUC values. The maximum improvement in AUC is observed for Ionosphere datasets that show up to a 12% increase in AUC of the compared reference techniques. On the other hand, a very small amount of improvement in AUC is observed for Car dataset. It should be noted that the datasets for which we did not get the significantly enhanced values, had already reached the results near to 1. To be precise, their results have already reached closer to the perfect classifier. Therefore, improving those results means designing a perfect classifier which seems to be practically difficult.

Imbalance Handling Method	**Haberman**	**Glass**	**Hepatitis**	**Ionosphere**	**Car**	**Hypothyroid**	**Pima**
RBB 1:1	0.710	0.958	0.859	0.832	0.9996	0.9993	0.832
RBB 3:1	0.693	0.954	0.853	0.823	0.9995	0.9995	0.823
UB 1:1	0.711	0.960	0.861	0.832	0.9997	0.9993	0.832
UB 3:1	0.694	0.952	0.854	0.824	0.9997	0.9994	0.824
Proposed OptDCE	0.741	0.963	0.889	0.923	0.9998	0.9996	0.835

Table 4. AUC of Different Imbalance Handling Techniques



(a)

(b)

(c)

(d)

(e)

(f)

(g)

Figure 7. AUC of Different Imbalance Handling Techniques

D. Comparison with Boosting-based Techniques

As mentioned in the previous section, the boosting-based ensembles are constructed sequentially wherein the next learner emphasizes the instances misclassified by the previous learner. Consequently, the generated classification model performs better than the bagging-based ensemble. On the other hand, bagging-based ensembles are faster than boosting-based ensembles. Therefore, the proposed OptDCE technique is compared with four existing boosting-based techniques by testing and accomplishing the classification of the various imbalanced datasets. The reference techniques selected for the comparison are as follows [22], [23]:

1. SMOTEBoost
2. DataBoost-IM
3. Ranked Minority Oversampling (RAMOBoost)
4. Re-sampling and AdaBoost-based Approach

E. Evaluation in terms of AUC

Table 5 given below represents the results of the proposed OptDCE technique in terms of AUC for various imbalanced datasets compared against the four reference techniques listed above.

Dataset	SMOTEBoost	DataBoostIM	RAMOBoost	Re-sampling and AdaBoost based Approach	Proposed OptDCE
Hepatitis	0.735	0.723	0.728	0.768	0.912
Ionosphere	0.824	0.844	0.825	0.863	0.911
Glass	0.931	0.917	0.937	0.956	0.963
Vehicle0	0.957	0.966	0.955	0.966	0.993
Vowel0	0.979	0.962	0.974	0.987	0.993
Car-vgood	0.999	0.992	0.971	0.993	0.997
Breast-W	0.960	0.944	0.950	0.960	0.991

Table 5. AUC of Different Imbalance Handling Techniques

The obtained results for the proposed OptDCE technique and four boosting-based CE in terms of AUC are illustrated in Figure 8. An analysis of the graphs indicates that the proposed OptDCE gives the highest AUC values for all the datasets. The performance wise ordering of the compared techniques indicates that the re-sampling and AdaBoost-based approach gives the second-highest AUC values. To be specific, up to a 26% increase in AUC values is recorded for the proposed classification approach to handle the imbalanced datasets. Especially, the achieved performance gains are very prominent for the Hepatitis dataset.

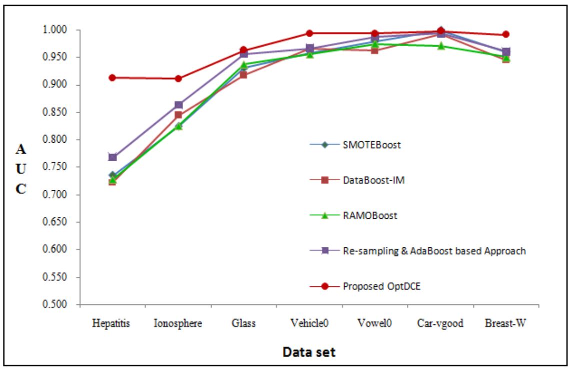


Figure 8. AUC of Different Imbalance Handling Techniques

V. Conclusion and Future Scope

The proposed classification algorithm is designed to deal with the class imbalance issue faced by many real-world applications. The first concern was to reduce the imbalance between the classes with the help of the re-sampling technique. Subsequently, the re-sampled training data with a reduced degree of imbalance is used to build the CE that is diverse in nature, optimal in size and performs significantly well.

A novel CE OptDCE improves the performance in classifying static imbalanced datasets by facilitating selection of the diverse and optimal set of mini ensembles to form the final prediction model that improves the classification performance without creating an unnecessarily larger ensemble. The validation of the proposed OptDCE against boosting-based approaches demonstrates the highest AUC values for the OptDCE technique. Up to a 26% increase in AUC value is recorded for the proposed OptDCE classification approach. For the class imbalance, the proposed OptDCE outperforms the compared bagging-based CEs giving greater AUC values. For example, Ionosphere datasets show up to a 12% increase in AUC of the compared reference techniques The research work can be further extended in order to deal with the class imbalance present in multiclass classification. Further, the work can be explored for handling the class imbalance issue in non-stationary environments as well. The presented diverse ensemble has been formed by using the pairwise diversity measure known as disagreement. Few other diversity measures can be explored in order to see their impact on the classifier performance.

References

[1] Herndon, Nic, and Doina Caragea. "A study of domain adaptation classifiers derived from logistic regression for the task of splice site prediction." IEEE transactions on nanobioscience 15, no. 2 (2016): 75-83.

[2] Kim, Kyounghoon, Helin Lin, Jin Young Choi, and Kiyoung Choi. "A design framework for hierarchical ensemble of multiple feature extractors and multiple classifiers." Pattern Recognition 52 (2016): 1-16.

[3] Yu, Hualong, and Jun Ni. "An improved ensemble learning method for classifying high-dimensional and imbalanced biomedicine data." IEEE/ACM transactions on computational biology and bioinformatics 11, no. 4 (2014): 657-666.

[4] Lee, Taehyung, Ki Bum Lee, and Chang Ouk Kim. "Performance of machine learning algorithms for class-imbalanced process fault detection problems." IEEE Transactions on Semiconductor Manufacturing 29, no. 4 (2016): 436-445.

[5] Abellán, Joaquín, and Javier G. Castellano. "A comparative study on base classifiers in ensemble methods for credit scoring." Expert Systems with Applications 73 (2017): 1-10.

[6] Sun, Yanmin, Mohamed S. Kamel, Andrew KC Wong, and Yang Wang. "Cost-sensitive boosting for classification of imbalanced data." Pattern Recognition 40, no. 12 (2007): 3358-3378.

[7] Bi, Wenjie, Meili Cai, Mengqi Liu, and Guo Li. "A big data clustering algorithm for mitigating the risk of customer churn." IEEE Transactions on Industrial Informatics 12, no. 3 (2016): 1270-1281.

[8] Amin, Adnan, Sajid Anwar, Awais Adnan, Muhammad Nawaz, Khalid Alawfi, Amir Hussain, and Kaizhu Huang. "Customer churn prediction in the telecommunication sector using a rough set approach." Neurocomputing 237 (2017): 242-254.

[9] Ahmed, Ammar AQ, and D. Maheswari. "Churn prediction on huge telecom data using hybrid firefly-based classification." Egyptian Informatics Journal 18, no. 3 (2017): 215-220.

[10] Xia, Xin, David Lo, Emad Shihab, and Xinyu Wang. "Automated bug report field reassignment and refinement prediction." IEEE Transactions on Reliability 65, no. 3 (2015): 1094-1113.

[11] Aburomman, Abdulla Amin, and Mamun Bin Ibne Reaz. "A novel SVM-kNN-PSO ensemble method for intrusion detection system." Applied Soft Computing 38 (2016): 360-372.

[12] Liu, Zhen, Ruoyu Wang, and Ming Tao. "SmoteAdaNL: a learning method for network traffic classification." Journal of Ambient Intelligence and Humanized Computing 7, no. 1 (2016): 121-130.

[13] Bao, Lei, Cao Juan, Jintao Li, and Yongdong Zhang. "Boosted Near-miss Under-sampling on SVM ensembles for concept detection in large-scale imbalanced datasets." Neurocomputing 172 (2016): 198-206.

[14] Vinodhini, G., and R. M. Chandrasekaran. "A sampling-based sentiment mining approach for e-commerce applications." Information Processing & Management 53, no. 1 (2017): 223-236.

[15] Bushara, N., and Ajith Abraham. "Novel ensemble method for long term rainfall prediction." International Journal of Computer Information Systems and Industrial Management Applications 7, no. 1 (2015): 116-130.

[16] Zhu, Bing, Bart Baesens, and Seppe KLM vanden Broucke. "An empirical comparison of techniques for the class imbalance problem in churn prediction." Information sciences 408 (2017): 84-99.

[17] Cavalcanti, George DC, Luiz S. Oliveira, Thiago JM Moura, and Guilherme V. Carvalho. "Combining diversity measures for ensemble pruning." Pattern Recognition Letters 74 (2016): 38-45.

[18] Salunkhe, Uma R., and Suresh N. Mali. "A hybrid approach for class imbalance problem in customer churn prediction: A novel extension to under-sampling." International Journal of Intelligent Systems and Applications 11, no. 5 (2018): 71.

[19] Chawla, Nitesh V., Aleksandar Lazarevic, Lawrence O. Hall, and Kevin W. Bowyer. "SMOTEBoost: Improving prediction of the minority class in boosting." In European conference on principles of data mining and knowledge discovery, pp. 107-119. Springer, Berlin, Heidelberg, 2003.

[20] Jain, Anju, Saroj Ratnoo, and Dinesh Kumar. "A novel multi-objective genetic algorithm approach to address class imbalance for disease diagnosis." International Journal of Information Technology (2020): 1-16.

[21] Lynam, Adam David. "Prediction of Oestrus in Dairy Cows: An Application of Machine Learning to Skewed Data." PhD diss., The University of Waikato, 2009.

[22] Thanathamathee, Putthiporn, and Chidchanok Lursinsap. "Handling imbalanced data sets with synthetic boundary data generation using bootstrap re-sampling and AdaBoost techniques." Pattern Recognition Letters 34, no. 12 (2013): 1339-1347.

[23] Feng, Wei, Wenjiang Huang, and Jinchang Ren. "Class imbalance ensemble learning based on the margin theory." Applied Sciences 8, no. 5 (2018): 815.

Author Biographies

Uma R. Godase has completed her Master of Engineering in CSE-IT and Doctor of Philosophy (PhD) degree in Computer Engineering from the University of Pune. Currently she is working as an Associate Professor at International Institute of Information Technology, Pune, Maharashtra, India. She has 20 years of experience in teaching field. She has published articles in various international journals and conferences. Her area of interest includes Security in Networks and Machine Learning.

Darshan V. Medhane has obtained his Master of Engineering degree in Computer Networks from the University of Pune, India. He received his Doctor of Philosophy (PhD) degree in Computer Science and Engineering from the Vellore Institute of Technology, Vellore, India. Currently he is working as an Associate Professor and Head of the department of Computer Engineering, MVPS's KBT College of Engineering, Nashik, India. He has authored more than twenty publications in peer reviewed international journals and conferences. His areas of interest are security in wireless networks, quantum computational intelligent systems, evolutionary multi-objective optimization, machine learning and position monitoring system.