JOURNAL OF INFORMATION SCIENCE AND ENGINEERING 37, 1011-1023 (2021)
DOI: 10.6688/JISE.202109_37(5).0002

DDCO – Diversified Data Characteristic-based Oversampling for Imbalance Classification Problems

GILLALA REKHA AND V. KRISHNA REDDY
Department of the Computer Science and Engineering
Koneru Lakshmaiah Education Foundation
Hyderabad, 500075 India
E-mail: gillala.rekha@klh.edu.in; vkrishnareddy@kluniversity.in

Although several techniques have been designed to handle class imbalance problems at data pre-processing level, they still face the difficult of over-generalization due to noisy minority samples and the overlapping region around class boundaries. In this study, an improved minority samples generation is proposed called Diversified Data Characteristic based Oversampling (DDCO) technique established on the instance characteristics of each dimension in the data space. In order to cope with over-generalization and overlapping problem, an improved minority samples generation is proposed to locate the newly generated synthetic samples in the minority region without any penetration into the majority space. The data characteristics of each dimension is used to control the location of the newly generated samples in same region. The performance of the proposed model has been evaluated on 14 imbalanced datasets and compared with state-of-the-art methods like SMOTE, Borderline-SMOTE, ADASYN, MWMOTE using AUC, and F-Measure as the performance measures. The results indicate significant improvement over the state-of-the-art methods.

Keywords: class imbalance problems, data pre-processing, data sampling methods, oversampling, synthetic sample generation

1. INTRODUCTION

Data imbalance remains a challenge for traditional machine learning algorithms affecting the classification problems. It occurs with uneven class distribution, wherein one class with more number of samples called majority class overtake other class with less number of samples called minority class [1]. Accordingly, the learning algorithms bias toward the majority class samples while training the model. Data imbalance prevalent in major domains including cancer malignancy grading [2], software defect prediction [3], network traffic classification [4], stock trend prediction [5], disease prediction [6] and many more. In the literature, a large number of techniques has been proposed to reduce the effect of data imbalance during model training [7, 8, 9, 10, 11, 12, 13]. These techniques are broadly categorized into data-level or data pre-processing techniques and algorithm-level technique. In the former, the imbalanced data is transform to well balanced data prior to classification either by generating the new samples for minority class

Received Sepember 23, 2020; revised November 1, 2020; accepted November 9, 2020.
Communicated by Maria José Sousa.

called oversampling, or by removing the existing samples from majority class called undersampling. Whereas in latter, the existing algorithms are modified to better accommodate the imbalance nature of the data. The existing algorithms are alleviated to reduce the bias towards the majority class samples during learning process. In comparison to algorithm-level methods, data-level techniques are more general because they do not rely on any particular learning algorithms, and can be combined with different techniques, for example, active learning and ensemble approaches effectively. In data pre-processing, resampling methods are most powerful techniques for handling class imbalance problems. But the main contribution of these techniques are based on imbalance ratio and may lead to over-generalization problem and increase in the overlapping regions between classes boundary [14]. The main issue remain open and need to explored is the need to analyzed the internal structure of data based on the attribute characteristics. In many applications, the data imbalance problem is challenging and subject to effective research efforts. It has been observed that class imbalance problem not only attributed with unequal distribution of data but also to a variety of factors such as class overlapping, small disjuncts and noisy data. Many studies reported that oversampling usually perform better than undersampling [15, 16] and most of the oversampling techniques consider imbalance ratio, thus neglecting the data characteristics that could help in generating the effective synthetic samples [17, 18]. Taking into the just mentioned limitation, this paper provides an efficient data diversity oversampling method by considering the data characteristics of each dimension in order to generate the new samples in minority region.

In this work, we propose a Diversified Data Characteristic-Based Oversampling (DDCO), to oversample the minority class samples based on the input data characteristics. The proposed model is part of the oversampling technique, and the aim is to produce the successful synthetic samples by taking into consideration the variability of the data samples of each element. The proposed method of producing synthetic samples provides better and more accurate results compared to previous methods. The following is the contribution of this work:

- We introduce an efficient approach to overcoming over-generalization due to noisy minority samples and decreasing the overlapping region around class boundaries while generating the synthetic samples.

- We present an improved synthetic sample generation process called Diversified Data Characteristic-based Oversampling (DDCO) that is capable of generating new samples with in the minority region.

- We conduct experiments on 14 benchmark datasets to compare the performance of the proposed method with state-of-the-art oversampling approaches.

The experimental results show that the suggested approach is comparable, and in most cases it outperforms other state-of-the-art methods. The rest of this paper is organized as follows. Section 2 presents literature review about the imbalance class distribution and the method applied at data level techniques. Section 3 introduces the proposed model. Next, Section 4 summarizes the experimental settings and result analysis. Conclusions are finally drawn in Section 5.

2. LITERATURE REVIEW

One of the most simple and popular methods to ease the imbalanced data problem is data-level techniques. These techniques are also called as re-sampling techniques in which the training data is pre-processed in order to balance the data sets before forwarding to machine learning algorithm. The existing data pre-processing techniques can be classified as 1) oversampling 2) undersampling and 3) composite method [19]. In oversampling method, synthetic samples are generated for minority class to balance the dataset. In undersampling method, some of the majority samples are eliminated to balance the dataset. Finally, the combination of both oversampling and undersampling forms composite method. As our work is concise to data level techniques, the literature survey is presented for the same.

Kubat and Matwin [20] were first to propose the resampling methods for class imbalance problems. The standard resampling methods are Random Under-Sampling (RUS) and Random OverSampling (ROS) [21, 22, 23]. In RUS, the majority class samples are selected and eliminated from the dataset and in ROS, minority samples are selected randomly with replacement and added to the original dataset. The main drawback of ROS is it simply duplicates the same original minority class samples in the class distribution. Further, the generation of synthetic samples has come into existence and Chawla *et al*. [26] proposed most popular oversampling technique called Synthetic Minority Over-sampling TEchnique (SMOTE). In this technique, the synthetic minority samples are generated by performing interpolation between minority class samples and its k-nearest neighbors. The effective implementation of SMOTE for various applications has motivated many researchers to propose different approaches for synthetic data generation to counter the class imbalance problem. The variant SMOTE approaches includes borderline-SMOTE [25], safe-level-SMOTE [24], ADASYN [27] and MWMOTE [14]. Borderline-SMOTE [25] generates synthetic samples in the borderline region between the classes. The two variations of Broderline-SMOTE [25] varies in selecting the samples. The first variation generates new samples using boundary region sample and its nearest minority neighbor while in the second, the new samples are generated by considering boundary region samples and its nearest neighbor samples from the complete dataset. The author argued that the Broderline-SMOTE provided better results than the SMOTE. But the main drawback is, it may generate the synthetic samples in overlapping and noisy regions which will degrade the classifier performance. To overcome the said drawbacks of borderline-SMOTE, the author [24] proposed Safe Level-SMOTE (SL-SMOTE). SL-SMOTE considers the safe level minority samples while generating the synthetic samples. The main focus was to generate the new samples closer to safe region. However, some synthetic samples may still be placed in the majority region which may degrade the classifiers performance [17]. The problem of over-generalization in SMOTE is addressed by Adaptive Synthetic Sampling Approach (ADASYN) [27]. It generates the synthetic minority samples adaptively based on their distribution in the data space. The author [14] proposed Majority Weighted Over-sampling Technique (MWMOTE) to generate synthetic samples from the most difficult minority samples. The difficult minority samples are assigned with weights based on the distance from the nearest majority samples. The main idea is to generate the new samples inside the minority class region. These techniques mainly focus on the k nearest neighbor while generating the synthetic samples but it is difficult to estimate k-values.

Apart from this, over-sampling of noisy minority samples generates more noisy samples [28]. As shown in Fig. 1, the data samples in yellow color represents the majority class samples and samples in red color represents minority class samples. The two main difficult part of the data is when it falls into other boundary [17] and are represented as 1) borderline samples that are positioned in the area surrounding the borders of the minority and majority class; and 2) noisy samples are individual samples of one class found in the protected areas of another class. As earlier discussed, the oversampling procedure does not consider data space restrictions while generating the synthetic samples and may some time mistakenly placed in the majority region. The aforementioned data level approaches generates synthetic data only by considering the linear relationship between the data points and mainly ignores the data characteristics of the samples. In this work, we propose a new techniques to generate the synthetic data by considering the data characteristics in order to place the samples in minority region.

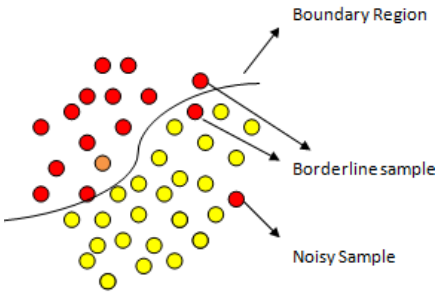


Fig. 1. Data samples with borderline and noisy samples on a decision boundary between two classes.

3. DDCO – DIVERSIFIED DATA CHARACTERISTIC-BASED OVERSAMPLING FOR IMBALANCE CLASSIFICATION PROBLEMS (PROPOSED METHOD)

This section introduces the proposed method, DDCO for imbalance classification problems, including motivation and the algorithm. The over-sampling methods mostly replicates the existing minority data leading to overlapping of samples in the feature space. Furthermore, the small disjunct problem and noisy data still exist and make the classification algorithm difficult to predict correctly the unseen data. The existing oversampling methods in the above mentioned section, tackle the above problem by generating synthetic samples to improve the generalization. However, SMOTE has been criticized for considering only minority class samples which leads to over generalization problem and may generate synthetic samples in majority region [14]. In this work, we propose a new technique to generate synthetic data by taking data characteristics into consideration. We first calculate the midpoint intervals by considering the data characteristics to provide diversity while generating synthetic samples. It is apparently capable of monitoring the generation of new samples accurately in the minority data space. The steps involved are defined as follows: For suppose, if the input dataset is D, the number of minority class

samples are represented as N_{min} where as the majority class samples are represented as N_{maj} respectively. Count variable holds the number of synthetic samples to be generated.

Step 1: Before starting the over-sampling we calculate the midpoints in the characteristic space for each dimension by considering the minimum and maximum values of each feature and dividing it by 2. The midpoints are denoted by midpts array:

midpts = $(midpt_1 + midpts_2, +, ..., +midpts_{ndim})$ where *ndim* is the number of features in the dataset. This step mainly focus on the diversity of the data.

Step 2: Next to compute the synthetic samples, we pick two samples minimum and maximum value with respect to its attribute values without replacement from minority sample set denoted as s_{min} and s_{max}.

Step 3: In the next step we compute the distance between the s_{max} and s_{min} for each attribute and store it in an array denoted as *dist*. To generate new sample, we used a random value denoted as rand which lie between (0.5,1] as been used in SMOTE to avoid overlapping of samples.

Step 4: Now, to place the new synthetic sample in the minority location. the new synthetic sample for each attribute is calculated as below:

$synsample_{att_i} = dist_i * rand + s_{min_i}$ if $(dist_i * rand_i + s_{min_i}) <= midpts_i$

else s_{max_i}-$(dist_i * rand)$

This over-sampling process is capable of generating the synthetic minority samples in the minority region and also prevent the samples fall in majority region.

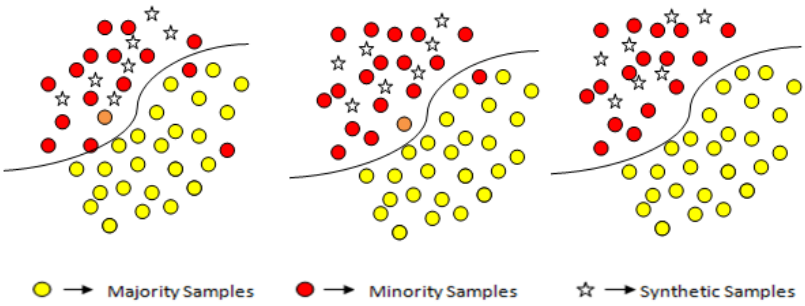


Fig. 2. Synthetic sample generation based on the proposed method.

As shown in Fig. 2, in all the three cases the newly generated samples are placed in the minority region only using the proposed method. The main goal of the proposed method is to generate synthetic samples by considering the characteristics of the attributes of the original data. In this work, we assume the characteristics of the features as important in synthetic data generation. Thus, the proposed algorithm provides efficient way of synthetic sample generation process without using *k*-nearest neighbor method as applied by almost all the oversampling methods.

Algorithm 1 shows the procedure for generating synthetic samples using data characteristics.

INPUT: Imbalanced Dataset 'N'
BEGIN
Split Dataset (N) into minority class and majority class
1: N_{min}, number of minority class samples
2: N_{maj}, number of majority class samples
// Consider Count variable to hold the number of new samples needed to balance the dataset
3: Count = $N_{maj} - N_{min}$
// Compute count no. of additional N_{min} samples
4: for $attr$ = 1 to nattr do :
Compute midpoints and save as midpts and
// take the midpoints of each attributes and store it in midpts
end for
// consider an array for holding the synthetic data
6: $newarr$ = 2d array of size $nattr$;
7: $lenmin$ = length of N_{min};
8: $lenmaj$ = length of N_{maj};
while ($lenmin$ is not equal to $lenmaj$):
$ns = emptyarray$;
for i = 0 to $nattr$ do:
copy elements of N_{min} into t;
from t remove rows t[i];
// Compute minimum value of ith column in Nmin and save it in mn
// Compute maximum value of ith column in Nmin and save it in mx
dist = $mx - mn$;
// calculate new sample
if (dist[i] * rand + mn) is lesser than or equal to midpts:
save dist[i] as new_{min}
else:
save $mx - midpts$ as new_{min} end else end if end for end while Return(new_{min});
END

Algorithm 1: Diversified Data Characteristic-based Oversampling method (DDCO)

4. EXPERIMENTS

In this section, we present the description of datasets used, the evaluation metric applied and the result analysis in detail and finally the results of the proposed method are compared with state-of-the-art methods like SMOTE [26], borderline-SMOTE [25], safe-level-SMOTE [24], and ADASYN [27].

4.1 Datasets

To evaluate the performance of the proposed model, we conducted experiments on fourteen datasets which are publicly available (https://sci2s.ugr.es/keel/imbalanced.php). The outline of the datasets are mentioned in Table 1, including the number of features, number of instances, size of minority samples and its imbalance ratio. Since we focus on the problem of binary classification which are similar to previous studies such as [7, 25, 26], so binary classification datasets were taken into consideration.

Table 1. Datasets used.

Name	No. of features	No. of Instances	No. of minority samples	Imbalance ratio
Abalone	8	731	42	16.4
Ecoli	7	336	52	5.46
Glass	9	214	70	2.06
Haberman	3	306	81	2.78
New-thyroid	5	215	35	5.14
Page-blocks	10	5472	559	8.79
Pima	8	768	268	1.87
Segmemt	19	2308	329	6.02
Shuttle	9	1829	123	13.87
Vehicle	18	846	199	3.25
Vowel	13	988	90	9.98
Wisconsin	9	683	239	1.86
Yeast	8	1484	429	2.46

4.2 Evaluation Metric

The most commonly used performance metrics for classification is accuracy. Accuracy is used to evaluate the conventional classification problems. On the contrary, accuracy is not sufficient for evaluating the imbalance classification problems and need to consider different aspects. However, various measures such as Area under the ROC curve (AUC), G-mean, F-measure proposed in the literature for class imbalance problems [29]. The ROC curve represents False Positive Rate (FPR) on x-axis and True Positive Rate (TPR) on y-axis. This measure provides single scalar value representing the performance of classification algorithm [30]. F-measure also called as f-score is the harmonic average of precision and recall. We employ F-measure and Area under Curve (ROC) for evaluating our proposed method. We considered two-class datasets for experiments. The majority class is the class with a larger number of instances and its smaller equivalent is the minority class. In our experiment, positive instance refer to minority class with value 1 and negative instance refer to majority class with value 0. The confusion matrix provides information about the actual and predicted values after classification. However, the classifier performance is evaluated based on the confusion matrix. Table 2 illustrates the confusion matrix for binary class problems. It comprises of four entries represented as True Positive (TP), True Negative (TN), False Positive (FP), False Negative (FN). The different performance metrics used in the experiment are provided in the Table 3.

Table 2. Confusion matrix.

		Predicted	
		Positive	Negative
Actual	Positive	TP	FN
	Negative	FP	TN

Table 3. Performance metrics with its formula.

Metric	Formula
Recall	$\frac{TP}{TP+FN}$
Precision	$\frac{TP}{TP+FP}$
Specificity	$\frac{TP}{TP+FN}$
Sensitivity	$\frac{TN}{TN+FP}$
F-measure	$\frac{2*Precision,\times Recall}{Precision+Recall}$
AUC	$\frac{Specificity+Sensitivity}{2}$

4.3 Results

The proposed model is applied on the real world data to balance them before training on the classification algorithms. The classification algorithms applied are Decision tree (C4.5) and Support Vector Machine (SVM). The efficiency of our model is compared against state-of-the-art methods like SMOTE, MWMOTE, borderline-SMOTE, safe-level SMOTE and ADASYN using AUC and F-measure metrics.

Table 4. F-measure using C4.5 classifier.

Dataset	SMOTE	Borderline-SMOTE	ADASYN	MWMOTE	Safe-level SMOTE	DDCO
Abalone	0.876	0.863	0.883	0.902	0.883	0.901
Ecoli	0.913	0.918	0.914	0.912	0.92	0.931
Glass	0.755	0.746	0.755	0.794	0.791	0.916
Haberman	0.643	0.641	0.628	0.661	0.637	0.667
Segment	0.742	0.736	0.702	0.73	0.735	0.745
Shuttle	0.858	0.854	0.86	0.885	0.864	0.892
Newthyroid	0.96	0.968	0.966	0.936	0.951	0.936
Page-blocks	0.958	0.959	0.958	0.961	0.949	0.984
Pima	0.686	0.7	0.684	0.686	0.682	0.791
Vehicle	0.675	0.693	0.657	0.672	0.685	0.857
Vowel	0.975	0.977	0.97	0.968	0.969	0.987
Wisconsin	0.941	0.942	0.932	0.947	0.945	0.965
Yeast	0.676	0.696	0.716	0.681	0.689	0.985

From the results we observed that, the proposed method is not sensitive to outliers existing in both the minority and the majority classes. In some datasets, like "Pima," "New- thyroid," the amount of outlier samples are very high. The results of F-measure and the AUC metric after oversampling the datasets using the proposed DDCO method and trained on C4.5and SVM classifier are presented in the last column of the results tables (refer Tables 4-7). Considering the obtained results of f-measure score and AUC

Table 5. AUC using C4.5 classifier.

Dataset	SMOTE	Borderline-SMOTE	ADASYN	MWMOTE	Safe-level SMOTE	DDCO
Abalone	0.679	0.713	0.675	0.61	0.671	0.645
Ecoli	0.851	0.841	0.868	0.846	0.868	0.878
Glass	0.735	0.737	0.744	0.782	0.784	0.925
Haberman	0.592	0.594	0.564	0.593	0.586	0.604
New-thyroid	0.738	0.732	0.698	0.729	0.735	0.762
Page-blocks	0.852	0.844	0.853	0.876	0.855	0.889
Pima	0.922	0.938	0.93	0.911	0.92	0.69
Segmemt	0.923	0.924	0.908	0.91	0.923	0.988
Shuttle	0.664	0.682	0.655	0.668	0.657	0.689
Vehicle	0.738	0.739	0.737	0.734	0.728	0.742
Vowel	0.928	0.929	0.901	0.912	0.927	0.944
Wisconsin	0.938	0.938	0.928	0.946	0.941	0.982
Yeast	0.825	0.828	0.843	0.809	0.822	0.988

Table 6. F-measure using SVM classifier.

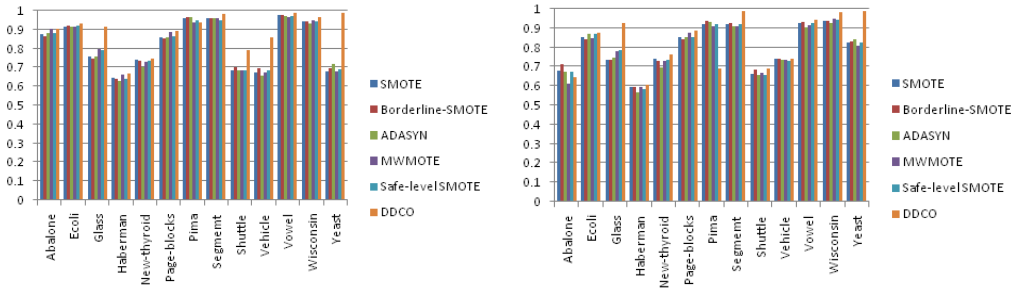
Dataset	SMOTE	Borderline-SMOTE	ADASYN	MWMOTE	Safe-level SMOTE	DDCO
Abalone	0.877	0.879	0.898	0.915	0.883	0.942
Ecoli	0.934	0.935	0.929	0.936	0.903	0.941
Glass	0.817	0.811	0.818	0.819	0.773	0.766
Haberman	0.754	0.767	0.757	0.778	0.716	0.789
New-thyroid	0.782	0.778	0.751	0.78	0.736	0.817
Page-blocks	0.909	0.896	0.892	0.885	0.864	0.906
Pima	0.959	0.959	0.96	0.943	0.953	0.968
Segmemt	0.967	0.969	0.978	0.976	0.951	0.945
Shuttle	0.735	0.725	0.732	0.731	0.694	0.738
Vehicle	0.77	0.76	0.766	0.767	0.717	0.857
Vowel	0.978	0.976	0.976	0.975	0.976	0.982
Wisconsin	0.965	0.962	0.961	0.955	0.938	0.974
Yeast	0.944	0.947	0.95	0.957	0.932	0.954

value, it is evident that by balancing the data using the proposed DDCO method, the classifier is able to classify both the minority and the majority samples on most test data sets.

The results are depicted in the graphical form as shown in Figs. 3 and 4. Fig. 3 presents the F-measure and AUC performance of the proposed method in comparison with state-of-the-art methods using C4.5 classifier. From the results (Figs. 3 (a) and (b)), it is clearly shown that out of 14 datasets, for 8 datasets the proposed DDCO method out performed and for the rest it is showing more or less the same performance as that of the state-of-the-art methods. Also, Fig. 4 presents the performance of DDCO method using SVM classifier. From the figure we observe that in most of the cases DDCO performed better than the other methods. Overall, the proposed method (DDCO) obtains the better results in terms of F-measure and AUC using SVM classification. Therefore, DDCO can provide an alternative solution to handle the class imbalance problem.

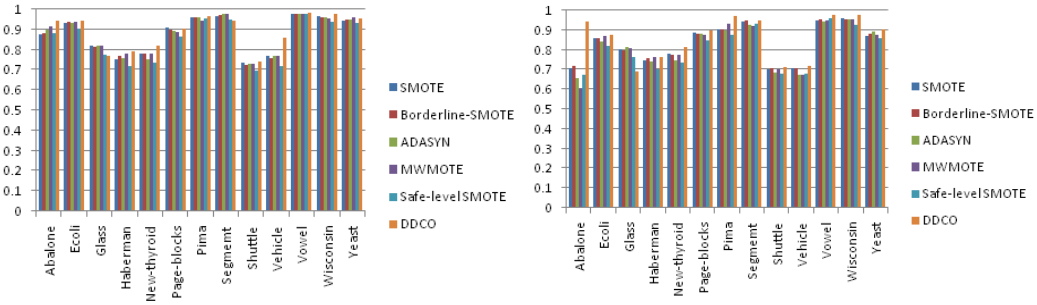
Table 7. AUC using SVM classifier.

Dataset	SMOTE	Borderline-SMOTE	ADASYN	MWMOTE	Safe-level SMOTE	DDCO
Abalone	0.707	0.718	0.656	0.608	0.672	0.944
Ecoli	0.861	0.859	0.841	0.867	0.82	0.873
Glass	0.801	0.795	0.814	0.809	0.761	0.69
Haberman	0.745	0.758	0.741	0.765	0.702	0.762
New-thyroid	0.78	0.772	0.744	0.775	0.734	0.812
Page-blocks	0.889	0.882	0.879	0.873	0.849	0.895
Pima	0.906	0.904	0.898	0.932	0.878	0.972
Segmemt	0.941	0.946	0.926	0.92	0.933	0.948
Shuttle	0.704	0.706	0.686	0.699	0.678	0.715
Vehicle	0.707	0.708	0.675	0.673	0.676	0.719
Vowel	0.947	0.955	0.945	0.947	0.96	0.979
Wisconsin	0.961	0.956	0.955	0.952	0.928	0.978
Yeast	0.871	0.881	0.891	0.874	0.859	0.902



(a) F-measure (b) AUC

Fig. 3. Performance results using C4.5 classifier.



(a) F-measure (b) AUC

Fig. 4. Performance results using SVM classifier.

5. CONCLUSION

This paper focus on the most significant shortcomings of oversampling 1) over generalization of minority samples and 2) overlapping between classes around the boundary regions. An improved technique, called Diversified Data Characteristic-based Oversampling for Imbalance Classification Problems (DDCO), was proposed which answers the two issues by taking data characteristics into consideration. The improved minority samples generation controls the location of the newly generated synthetic samples in the minority region. In the experiments, 14 datasets with different imbalance ratios were utilized to evaluate the performance of the proposed method. Decision tree (C4.5) and SVM classifier were trained on the acquired datasets after pre-processing and tested using F-measure and AUC metric. The results of the models were compared with state-of-the-art methods. The results demonstrated that the proposed DDCO technique has a better performance, particularly when the dataset is highly imbalanced. As future direction, it is important to examine the proposed DDCO model on multi-class imbalance problems.

REFERENCES

1. S. R. Searle and S. R. Searle, *Linear Models for Unbalanced Data*, Vol. 1987, Wiley, NY, 1987.
2. S. Liu, J. Zhang, Y. Xiang, W. L. Zhou, D. X. Xiang, "A study of data pre-processing techniques for imbalanced biomedical data classification," *arXiv Preprint*, 2019, arXiv:1911.00996.
3. S. Wang and X. Yao, "Using class imbalance learning for software defect prediction," *IEEE Transactions on Reliability*, Vol. 62, 2013, pp. 434-443.
4. S. E. Gómez, L. Hernández-Callejo, B. Martínez, A. J. Sánchez-Esguevillas, "Exploratory study on class imbalance and solutions for network traffic classification," *Neurocomputing*, Vol. 343, 2019, pp. 100-119.
5. P. Rajesh, N. Srinivas, K. V. Reddy, G. VamsiPriya, M. V. Dwija, and D. Himaja, "Stock trend prediction using ensemble learning techniques in python," *International Journal of Innovative Technology and Exploring Engineering*, Vol. 8, 2019, pp. 150-155.
6. T. Sajana and M. R. Narasingarao, "Classification of imbalanced Malaria disease using naïve bayesian algorithm," *International Journal of Engineering & Technology*, Vol. 7, 2018, pp. 786-790.
7. J. A. Sáez, J. Luengo, J. Stefanowski, F. Herrera, "SMOTE-IPF: Addressing the noisy and borderline examples problem in imbalanced classification by a re-sampling method with filtering," *Information Sciences*, 2015, Vol. 291, pp. 184-203.
8. L. A. Bewoor, V. C. Prakash, and S. U. Sapkal, "Evolutionary hybrid particle swarm optimization algorithm for solving NP-hard no-wait flow shop scheduling problems," *Algorithms*, Vol. 10, 2017, p. 121.
9. C. Amarendra and K. H. Reddy, "PSO algorithm support switching pulse sequence ISVM for six-phase matrix converter-fed drives," *Smart Intelligent Computing and Applications*, 2019, pp. 559-569.

10. N. Namassivaya, S. Pal, and D. V. Ratnam, "Modelling of FPGA-particle swarm optimized GNSS receiver for satellite applications," *Wireless Personal Communications*, Vol. 106, 2019, pp. 879-895.
11. S. P. Potharaju and M. Sreedevi, "A novel LtR and RtL framework for subset feature selection (Reduction) for improving the classification accuracy," in *Progress in Advanced Computing and Intelligent Engineering*, 2019, pp. 215-224.
12. K. Thirugnanasambandam, S. Prakash, V. Subramanian, S. Pothula, and V. Thirumal, "Reinforced cuckoo search algorithm-based multimodal optimization," *Applied Intelligence*, Vol. 49, 2019, pp. 2059-2083.
13. K. A. Sultanpure and L. S. S. Reddy, "Job scheduling for energy efficiency using artificial bee colony through virtualization," *International Journal of Intelligent Engineering and Systems*, Vol. 11, 2018, pp. 138-148.
14. S. Barua, M. Islam, X. Yao, and K. Murase, "MWMOTE-majority weighted minority oversampling technique for imbalanced data set learning," *IEEE Transactions on Knowledge and Data Engineering*, Vol. 26, 2012, pp. 405-425.
15. B. W. Yap, K. Abd Rani, H. A. Abd Rahman, S. Fong, Z. Khairudin, and N. N. Abdullah, "An application of oversampling, undersampling, bagging and boosting in handling imbalanced datasets," in *Proceedings of the 1st International Conference on Advanced Data and Information Engineering*, 2014, pp. 13-22.
16. G. Rekha, A. K. Tyagi, and V. K. Reddy, "Solving class imbalance problem using bagging, boosting techniques, with and without using noise filtering method," *International Journal of Hybrid Intelligent Systems*, Vol. 15, 2019, pp. 67-76.
17. K. Napierala and J. Stefanowski, "Types of minority class examples and their influence on learning classifiers from imbalanced data," *Journal of Intelligent Information Systems*, Vol. 46, 2016, pp. 563-597.
18. G. Rekha and V. K. Reddy, "A novel approach for handling outliers in imbalance data," *International Journal of Engineering & Technology*, Vol. 7, 2018, pp. 1-5.
19. G. Rekha, A. K. Tyagi, and V. K. Reddy, "A wide scale classification of class imbalance problem and its solutions: A systematic literature review," *Journal of Computer Science*, Vol. 15, 2019, pp. 886-929.
20. M. Kubat, S. Matwin, *et al*., "Addressing the curse of imbalanced training sets: one-sided selection," *Citeseer*, Vol. 97, 1997, pp. 179-186.
21. S. Kotsiantis, D. Kanellopoulos, P. Pintelas, *et al*., "Handling imbalanced datasets: A review," *GESTS International Transactions on Computer Science and Engineering*, Vol. 30, 2006, pp. 25-36.
22. N. Japkowicz and S. Stephen, "The class imbalance problem: A systematic study," *Intelligent Data Analysis*, Vol. 6, 2002, pp. 429-449.
23. G. Rekha and A. K. Tyagi, "Necessary information to know to solve class imbalance problem: From a user's perspective," in *Proceedings of the 2nd International Conference on Recent Innovations in Computing*, 2020, pp. 645-658.
24. C. Bunkhumpornpat, K. Sinapiromsaran, and C. Lursinsap, "Safe-level-smote: Safe-level-synthetic minority over-sampling technique for handling the class imbalanced problem," in *Proceedings of Pacific-Asia Conference on Knowledge Discovery and Data Mining*, 2009, pp. 475-482.

25. H. Han, W. Y. Wang, and B. H. Mao, "Borderline-SMOTE: a new over-sampling method in imbalanced data sets learning," in *Proceedings of International Conference on Intelligent Computing*, 2005, pp. 878-887.
26. 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, 2002, pp. 321-357.
27. H. He, Y. Bai, E. A. Garcia, and S. Li, "ADASYN: Adaptive synthetic sampling approach for imbalanced learning," in *Proceedings of IEEE International Joint Conference on Neural Networks*, 2008, pp. 1322-1328.
28. H. He and E. A. Garcia, "Learning from imbalanced data," *IEEE Transactions on Knowledge and Data Engineering*, Vol. 21, 2009, pp. 1263-1284.
29. M. Hossin and M. Sulaiman, "A review on evaluation metrics for data classification evaluations," *International Journal of Data Mining & Knowledge Management Process*, Vol. 5, 2015, p. 1.
30. A. P. Bradley, "The use of the area under the ROC curve in the evaluation of machine learning algorithms," *Pattern Recognition*, Vol. 30, 1997, pp. 1145-1159.

Gillala Rekha received her M.Tech degree in Computer Science and Engineering from Jawaharlal Nehru Technological University, India, in 2009. Currently she is pursuing Ph.D. at Department of Computer Science and Engineering of Koneru Lakshmaiah University and is a member of the CSI. Her research interests are in machine learning and data mining, especially in pre-processing, ensemble learning and class imbalance learning, big data.

V. Krishna Reddy received his Ph.D. from Acharya Nagarjuna University. Currently, he is holding the position of Principal FED, KL University. His area of interests are in machine learning, data mining, cloud computing.