Detection of Class Imbalance Problem using Oversampling Technique

Revati V. Mundle[1] M. S. Chaudhari[2]
[1,2]Department of Computer Science & Engineering
[1,2]PBCOE, Nagpur, India

Abstract— Now a day as the application area of technology increases the proportion of data and the nature also increases. One of such problem in data mining and machine learning techniques are class imbalance problem. Class imbalance problem is a problem where distribution of data largely belongs to one class while small or no data belongs to other class. Class imbalance problem is also known as skewed data problem. The data in real-world applications often has imbalanced class distribution where most of the classifier correctly classifies majority class data while they completely ignore minority. This is the problem associated with class imbalance problem. There many techniques used to solve class imbalance problem such data preprocessing, algorithmic approach and ensemble techniques. Data preprocessing gives better solution than other techniques. Data preprocessing techniques broadly classified into oversampling and under sampling technique. The disadvantage associated with under sampling is that it losses the information. So, mostly oversampling technique is used to balance the data. But disadvantage associated with the oversampling techniques is that it replicates unnecessary information. In this paper, proposed an approach to minimize the problem of replication of data associated with the oversampling technique.

Key words: Imbalance Class Distribution, Class Imbalance Problem, Under-Sampling, Oversampling

I. INTRODUCTION

In many real time applications large amount of data is generated with skewed distribution. A data set said to be highly skewed or class imbalanced if sample from one class is in higher number than other. In imbalance data set the class having more number of instances is called as major class while the one having relatively less number of Instances are called as minor class [1]. There are a large number of real-world applications that give rise to data sets with an imbalance between the classes. Examples of these kinds of application include medical diagnosis, oil spill detection [2], fraud detection [3], text classification [4], as medical diagnosis detection of rare but important disease is very important than regular treatment. Similar situations are observed in other areas, such as detecting fraud transaction in banking operations [3], detecting network intrusions, managing risk and predicting failures of technical equipment. The imbalance of class data set can be severe. In case of small sample data sets, such as those with hundreds of samples or less, the class can have 1 minority sample to 10 or 20 majority samples. In larger data sets that contain multiple thousands of samples, the class imbalance may be even larger; some data sets have an imbalance ratio of 1 minority sample to 100, 1000, and even 10000 majority samples, which is worse [6]. As the skew increases, performance noticeably drops on the minority class.

The classification techniques usually consider a balanced class distribution (i.e. there data in the class is equally distributed). Usually, a classifier performs well when the classification technique is applied to a dataset evenly or equally distributed to different classes. Still many real applications face the imbalanced class distribution problem. In such situation, the classification task imposes difficulties when the classes present in the training data are imbalanced. The problem of imbalanced class distribution occurs when one class is represented by a large number of examples (majority class) while the remaining other is represented by only a few (minority class). In this case, a classifier usually tends to predict that samples belong to majority class and completely ignore the minority class. This is known as the class imbalance problem. Fig 1.1 illustrates the idea of the class imbalance problem where a minority class is represented by only 1% of the training data and 99% for majority class.

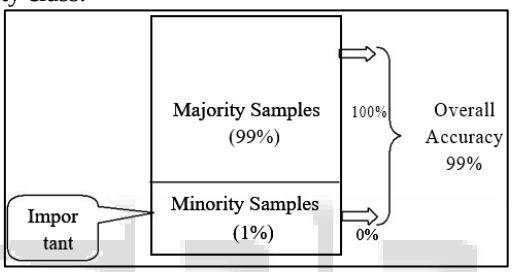

Fig. 1: Distribution of Majority and Minority class

In Class Imbalance, Problem are raised when one class having more samples than other classes. A dataset is imbalanced if the classification categories are not comparatively equally represented. The level of imbalance (ratio of size of the majority class to minority class) can be as vast as 1:99. It is noteworthy that class imbalance is emerging as an important issue in designing classifiers. The classical classifiers of balance datasets cannot deal with the class-imbalance problem because they pay more attention to the majority class. The main drawback associated with majority class is loss of important information. The class imbalance problem is a difficult challenge faced by machine learning and data mining. In the last ten years it has attracted a significant amount of research. A classifi er affected by the class imbalance problem for a specific data set would see strong accuracy overall but very poor performance on the minority class [6]. The problem can appear in two types of data sets:

1) Binary problems where one of the two classes is composed of considerably more number samples than the other class, and
2) Multi-class problems where each class only contains a small fraction of the samples and use one-versus-rest classifiers.

Data sets meeting one of the two above criteria have different misclassification costs for the different classes. The costs for classifying samples into different classes are given in a cost matrix. The definitive cost matrix for a problem is occasionally explicitly stated, but many of the time, it is simply an inherent part of the problem. Thus, an algorithm will either have to determine the best cost matrix during

All rights reserved by www.ijsrd.com

training [5], or the user will have to select a cost matrix to use in training. If the chosen cost matrix is incorrect, it can lead to imperfect decisions from the classifier, so it is extremely important when doing cost-sensitive learning that the proper cost matrix be used [7].

II. PERFORMANCE METRICS

It is a common to represent the accuracy of classification using confusion matrix. The columns are the Predicted class and the rows are the Actual class. In the confusion matrix, TN is the number of negative examples correctly classified (True Negatives), FP is the number of negative examples incorrectly classified as positive (False Positives), FN is the number of positive examples incorrectly classified as negative (False Negatives) and TP is the number of positive examples correctly classified as positive (True Positives). The Predicted accuracy is defined as Accuracy = (TP+ TN) / (TP+FP+TN+FN)

Class Name	Predicted Positive Class	Predicted Negative Class
Actual Positive Class	TP (True Positive)	FN (False Negative)
Actual Negative Class	FP (False Positive)	TN (True Negative)

Table 1: Confusion Matrix

The predictive accuracy may not be appropriate when the data imbalanced and/or the costs of different errors vary markedly. Precision, Recall, Specificity, F-measure and G -mean can be computed from confusion matrix as specified in equations (1),(2),(3) and (4).AUROC (Area Under Receiver Operating Characteristics Curve) is in general used to evaluate the performance of a binary classifier.

$$\text{Precision}=TP / (TP+FP) \quad (1)$$

$$\text{Recall}=TPR= TP / (TP+FN) \quad (2)$$

$$\text{Specificity}=TN / (TN+FP) \quad (3)$$

$$\text{F-measure}= (2*\text{Recall}*\text{Precision}) / (\text{Recall}+\text{Precision}) \quad (4)$$

$$\text{G-mean}= \sqrt{\text{Recall} * \text{Specificity}} \quad (5)$$

$$FPR= FP / (FP+TN) \quad (6)$$

Recall measures the predicted accuracy of the positive samples (minority samples). Precision refers to the ratio of actual positive samples to all samples that are predicted as being positive while Recall is the ratio of actual positive samples to all samples of actual positive samples that are correctly identified by the classifier, which is the same as Sensitivity. Mostly for a classifier, the recall rate will be low if the precision rate is high, that is, the two criteria are trade - off. For unbalanced data sets, higher the recall lowers the precision. So increasing recall rates without decreasing the precision of the minority class is a challenging problem. F-Measure is a popular measure for unbalanced data classification problems [8]. F-Measure depicts the trade-off between precision and recall. The geometric mean (GM) [9] simultaneously maximizes the accuracy in positive and negative examples with a favorable trade-off. So use G-mean to maximize accuracy of majority samples and Recall with a favorable trade-off. ROC Curves are drawn by plotting FPR on X- axis and TPR on Y-axis for various thresholds. The area under ROC curve is a good measure of prediction accuracy.

III. RELATED WORK

The imbalanced data problem in classification can appear in two different types of data sets: binary problems, where one class having more number of samples than the other and multi-class problems, where the applications have more than two classes and unbalanced class distribution hinder the classification performance. In order to overcome the class imbalance problem, many approaches have been introduced. Most research efforts on imbalanced data sets have traditionally concentrated on two-class problems.

Two class problem or binary class problem occurs when there are significantly fewer training instances of the first class compared to other one [12]. For example, in credit card usage data there are very few cases of fraud transactions as compared to the number of normal transaction. So, the instances of this data set belong to either fraud class or normal class only. For two-class problem, researcher proposed many solutions to the class imbalance problem at both data level and algorithm level. In data level different re- sampling techniques are applied to balance class distribution, such as resampling techniques and multi classifier committee approach [22]. In algorithm level solutions try to adapt existing classifier learning algorithms to strengthen learning with regards to the small class, such as recognition based learning, ensemble learning, and cost-sensitive learning data set to obtain a more balanced number of instances in each class. To minimize class imbalance in training data, there are two basic methods, under sampling and over sampling [7], [8].

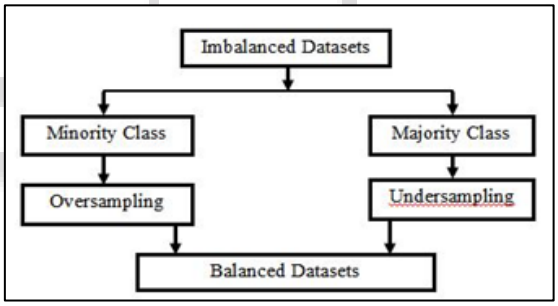

Fig. 2: Handling Dataset

A. *Undersampling*

It removes data from the original data set by randomly select a set of majority class example and then remove this sample [12]. Hence, an under sample approach is aim to decrease the skewed distribution of majority class and minority class by lowering the size of majority class [8]. Under-sampling is suitable for large application where the number of majority samples is very large and lessening the training instances reduces the training time and storage [15]. Fig 3.2 illustrates the distribution of samples in a dataset before and after applies undersample approach. For example, from the Fig 3.2 we find the red circle is represent minority class which has two instances. So, for this reason we take randomly only two instances from other circles: blue which represent majority classes in this case. The drawback of this technique is that there does not exist a control to remove patterns of the majority class, thus it can discard data potentially important for the classification process [13], which degrade classifier performance.

All rights reserved by *www.ijsrd.com*

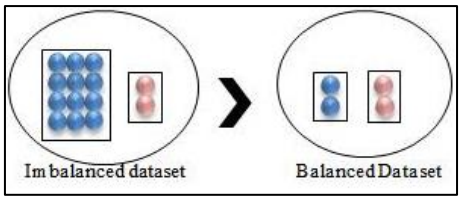

Fig. 3: The distributions of samples before and after under sample approach.

B. Oversampling

It is a method to adding a set of sampled from minority class by randomly select minority class examples and then replicating the selected examples and adding them to data set [18]. The advantage is that no information is lost, all instances are employed. However, the major problem of this technique is leads to a higher computational cost. Fig 3.3 illustrates the distribution of samples in a dataset before and after applies over sample approach. For example, from the Fig. nine instances. So, for this reason we replicate instances from other circles: red which represents minority classes until they reach to nine instances approximately in this case. The drawback of this technique is if some of the small class samples contain labeling error, adding them will actually deteriorate the classification performance on the small class [21].

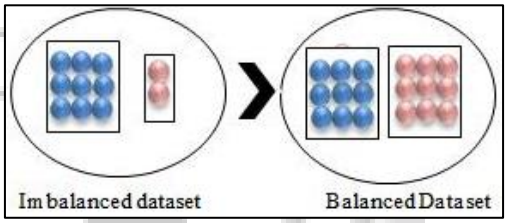

Fig. 4: The distributions of samples before and after over sample approach.

We find the blue circle represents majority class which has However, both oversampling and under sampling are capable of solving the imbalance class problem and both of them having their own advantage and disadvantage. Comparing oversampling and under resampling, observation simply favoring oversampling is that under-sampling removes some data from the original data, that data may be important so it result in loss of information while oversampling does not suffer from this problem.

Hybrid Method: In this method, both oversampling and under sampling method is used to balance the dataset. In hybrid method, oversampling is usually done on minority class samples and under sampling is usually done on majority class samples [19]. Hybrid method can also utilize using bagging and boosting [24].

Random Under sampling (RUS): Random oversampling balances a data set by duplicating examples of the minority class until a desired class ratio is achieved. The benefit of using undersampling to balance the class distribution in this data set is that the time required to train the classifier will be relatively short, but a smaller training data set also has its drawbacks that is loss of valuable information [13].

Random Oversampling (ROS): The most important method in under- sampling is random under-sampling method which trying to balance the distribution of class by randomly removing majority class sample. Random under sampling (RUS) removes examples (randomly) from the majority class until the desired balance is achieved. It has the benefit, however, of decreasing the time required to train the models since the training data set size is reduced[11],[14].

IV. Proposed System

A. Proposed Approach

The proposed approach is based on the idea of using the probability distribution of minority class to generate new minority class training samples. This way can avoid the possibility of the synthetically generating training samples actually belonging to any other class in case of class overlap. In order to sample from the probability distribution of the minority class, first need to estimate the distribution and then employ a mechanism to generate samples from that distribution. In the following, apply Oversampling approach for strategically selecting minority class [16] data sample that have the highest probability of being misclassified by existing learning model and generating new minority class sample to balance the imbalance distribution of data.

B. Proposed Architecture

In this architecture create a classifier based on the input dataset. Run classifier and classify the dataset in binary class. In classification, decision tree classifier [17] builds a model on training data and checks it with test data. The model is built with decision tree [14] classifier which deploys the classification task and gives correctly and incorrectly classified data. Then dataset is given to Clustering. In k- nn clustering, dataset is clustered [18] in such a way that it clusters misclassified data. Select only those clusters which have minority samples that are misclassified. Then selected minority samples are given to probabilistic oversampling technique. Dataset which is obtained after by applying oversampling gives back to check whether the data set is balanced or not. The probabilistic oversampling technique [1], [12], [13] handles the imbalanced datasets to get balanced datasets.

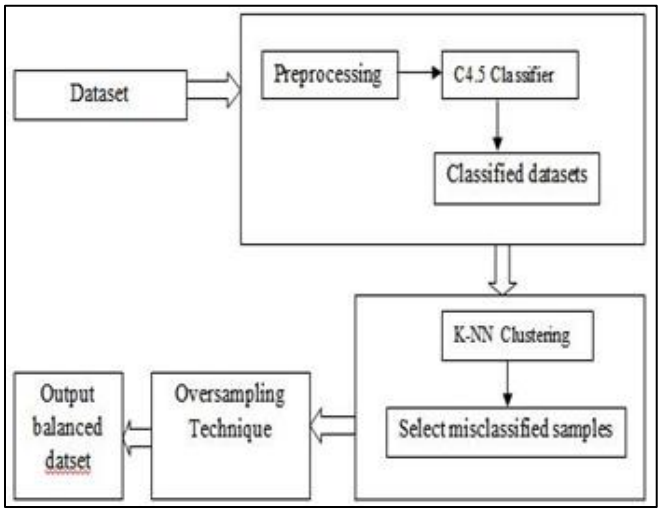

Fig. 5:Proposed Architecture

C. Verify Dataset

To check whether dataset is balanced or not process through verify stage [21].In this we get the count of the majority and minority samples which gives result as balanced or imbalanced dataset. Here, Haberman's Survival Data is input dataset where class labelled 1 represent the patient survived 5

All rights reserved by www.ijsrd.com

years or longer and class labelled 2 represents the patient died within 5 year.If dataset is imbalanced it is given to the classifier.

1) Classification

Build classifier model using C4.5 decision tree classifer [23],[24],[25] where the model is build using training dataset and verify using test dataset. The decision tree [20] generated from a set of training data by C4.5, using the concept of information entropy. At each node of the tree, C4.5 chooses one attribute of the data that most effectively splits. The criterion is the normalized information gain which is result from choosing an attribute for splitting the data. So, the attribute which has the highest normalized information gain is chosen to make the decision tree. Decision tree algorithm [15] was constructed in a top-down recursive divide-and-conquer manner.

2) Clustering

The goal of this clustering method [16] is to simply seperate the data based on the assumed similarties between various classes. Thus, the classes can be differentiated from one another by searching for similarities between the data provided. A distance is assigned between all points in a dataset. Distance is defined by the Euclidean distance between two points or:

$$d = \sqrt{\sum_{i=0}^{i=n} (x_i - y_i)^2}$$

From these distances, a distance matrix is constructed between all possible pairings of points (x, y). Each data point within thedata set has a class label in the set, C={c1,...,cn}. The data points', k-closest neighbors (k being the number of neighbors) are then found by analyzing the distance matrix. The k-closest data points are then analyzed to determine which class label is the most common among the set. The most common class label is then assigned to the data point being analysed. So, three clusters are generated as result where first cluster with samples of class 1, second with samples of class 2 and third with samples having both class which is misclassified samples.

Next select the samples from misclassified cluster and do the oversampling on the minority class samples which helps to balance the dataset.

V. RESULTS

The Dataset is first need to check whether is balanced or not. If dataset is balanced then no need to do further process as proposed work is to balance the dataset. The dataset is said to be imbalance if one class contains more number of samples then other which is given in Fig 4.1where it process the dataset to verify whether it is balanced or not. The Imbalanced Dataset is given to the classifier.

This chapter present and analyse implemented results. In the first section, accuracy is calculated and comparison is given before and after proposed method. In the second section, other performance metrices are calculated and comparison is given. Finally, discussed the results obtained and summarized the results.

A. Accuracy of Classifier

In this section, classify instances before doing any change on data sets to compare the classification performance of classifier before using proposed method and after using it. Classification performance metrices before oversampling are used as a baseline.

Table 1 illustrates the average accuracy for all the dataset before and after oversampling. Accuracy before oversampling is computed without doing any process on dataset. Accuracy after oversampling is computed by applying KNN based minority data selection on each data set to know if there is any improvement happens. In general, oversampling has many advantages over under sampling approach but the disadvantage of oversampling is that it replicates unnecessary data. To avoid this disadvantage apply proposed KNN based minority oversampling technique which balance the data and classify data more accurately. Note that the average accuracy range improves from 53 to 93 for haberman dataset which is considered as a good result. However; it cannot depend only on accuracy metric as a measure for classification for imbalanced data as mentioned in chapter 3. Therefore, computed other performance factor of a classifier. Compute the overall accuracy for all the data set as illustrated in Table 2.

Dataset	Accuracy before oversampling	Accuracy after oversampling
Haberman	53.08	93.82
Ecoli	76	88
Shuttle	66	96

Table 2: Accuracy of classifier before and after oversampling

Figure 1 shows graph of the average accuracy before and after the oversampling in graphical form. The graph shows that the proposed oversampling approach readily improves the accuracy of each dataset.

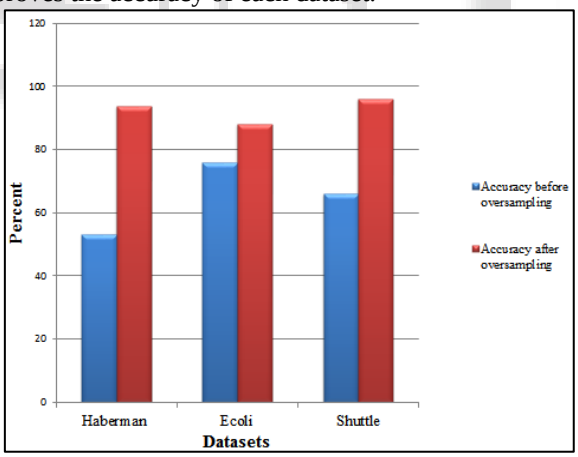

Fig. 6: Accuracy graph before and after oversampling

B. Performance Parameter of Classifier

Table 3 shows the performance parameter of classifier before oversampling. Note that in general the average of F-measure is less than the average of accuracy that it means the data sets have imbalance problem. So, the accuracy measure cannot detect the imbalanced problem and cannot give us the actual classification performance especially when the data has imbalanced class distribution problem.

Dataset Name	Sensitivity	Precision	Specificity	F-measure	G-Mean
Haberman	1.0	0.48	0.04	0.64	0.21

All rights reserved by *www.ijsrd.com*

Ecoli	0.76	1.0	0	0.86	0
Shuttle	0.87	1.0	1.0	0.93	0.95
Haberman	1.0	0.82	1.0	0.81	0.90
Ecoli	1	0.93	0.8	0.96	0.89
Shuttle	0.83	1.0	1.0	0.92	0.90

Table 3: Classifier performance parameter before oversampling

Table 3 shows the performance parameter of classifier after oversampling. Note that the performance metrics are also improved after the proposed approach of oversampling. Comparing the performance parameter for haberman shows that the sensitivity is maintained to 1.0, precision improves from 0.48 to 0.82, specificity is improved 0.04 to1.0, f-measure improves from 0.64 to 0.81 and g-mean improves from 0.21 to 0.90.

Figure 7 is graph of performance of classifier before oversampling given by using the values computed before oversampling. The graph shows the performance parameter of three datasets.

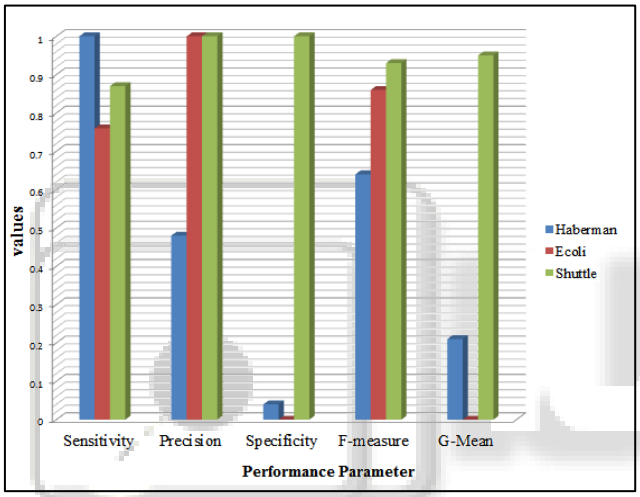

Fig. 7: Performance parameter of classifier before oversampling

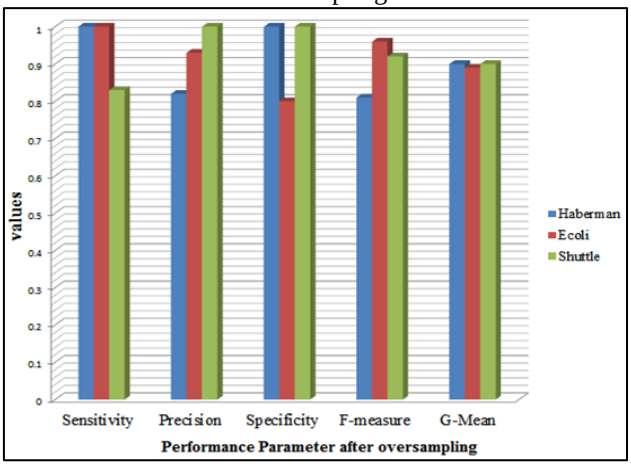

Fig. 8: Performance parameter of classifier after oversampling

Figure 7 is graph of performance parameter of classifier after oversampling given by using the values computed after oversampling. The graph shows the performance parameter of three datasets. Comparing the graph of performance parameter before and after oversampling readily shows that the parameter values are maintained or improved.

VI. Summary

The overall accuracy after applying approach on imbalance dataset is improved as compare to the accuracy before applying approach. The accuracy of Haberman, Ecoli, and Shuttle is increased up to 30 to 40 % after applying approach. Table 2 and Table 3 illustrate the improvement in accuracy and other parameter after applying approach on imbalanced dataset. Note the great difference in improvement before processing and after applying approach in accuracy and F-measure. From all the above results confirm minority oversampling approach improves the accuracy and minimize the misclassification rate of classifier.

References

[1] Barman Das, Narayanan C. Krishnan, and Diane J. Cook, "Racog and Wracog Two Probabilistic Oversampling Techniques" IEEE Transactions on Knowledge and Data Engineering, Vol. 27, No. 1, January 2015, Pp 222-232.

[2] M. Kubat, R. Holte, and S. Matwin, "Machine learning for the detection of oil spills in satellite radar images," Mach. Learn., vol. 30, no. 2, pp. 195–215, 1998

[3] Barandela. R, Sanchez. J, S. Garc´a. V, and Rangel. E, ―Strategies for learning in class imbalance problems, Pattern Recognition, 2003, pp. 849– 851.

[4] Wattana Jindaluang and Varin Chouvatut,Sanpawat Kantabutra "Under-Sampling By Algorithm With Performance Guaranteed For Class-Imbalance Problem" 2014 International Computer Science And Engineering Conference (ICSEC), Pp 812-823.

[5] He, H., and Garcia, E.: Learning from imbalanced data. IEEE Transactions on Knowledge and Data Engineering 21(9), 1263 –1284, 2009.

[6] Shuo Wang, Member, and Xin Yao, "Multiclass Imbalance Problems: Analysis and Potential Solutions", IEEE Transactions on Systems, Man, and Cybernetics—Part B: Cybernetics, Vol. 42, No. 4, August 2012.

[7] QiangWang, "Research Article on A Hybrid Sampling SVM Approach to Imbalanced Data Classification", Hindawi Publishing Corporation Abstract and Applied Analysis Volume 2014, Article ID 972786, 7 pages http://dx.doi.org/10.1155/2014/972786.

[8] Shuo Wang, Member, Xin Yao, "Multiclass Imbalance Problems: Anlysis and Potential Solutions" IEEE Transactions on Systems, Man, and Cybernetics—Part B: Cybernetics, Vol. 42, No. 4, August 2012, Pp 1356-1359.

[9] Rushi Longadge, Snehlata S. Dongre, Latesh Malik "Multi-Cluster Based Approach for Skewed Data in Data Mining" E-ISSN: 2278-0661, P- ISSN: 2278-8727volume 12, Issue 6 (Jul. - Aug. 2013), Pp 66-73.

[10] G. Weiss and F. J. Provost. Learning when training data are costly: The effect of class distribution on tree induction. J. Artif. Intell. Res.(JAIR) 19 : 315-354, 2003.

[11] J. Davis and M. Goadrich. The relationship between precision-recall and ROC curves. In Proc. of the 23rd International Conference on Machine Learning, pp 233-240, 2006.

[12] L. Breiman. Bagging predictors. *Machine Learning*, 24(2):123-140, 1996.

[13] L. Breiman. Random forests. Machine Learning, 45(1): 5-32, 2001. R.N. Lichtnwalter and N. V. Chawla. Link

All rights reserved by *www.ijsrd.com*

prediction: fair and effective evaluation. Advances in Social Networks Analysis and Mining (ASONAM), IEEE/ACM International Conference on. IEEE, pp:376-383, 2012.

[14] L. Breiman, Bagging predictors, Machine Learning 24(2), pp: 123-140,1996Jia Li, Hui Li, " Jun-Ling Yu, ―Application Of Random-Smote On Imbalanced Data Mining", 2011 Fourth International Conference On Business Intelligence And Financial Engineering.

[15] Susana Barua, Md. Monirul Islam, Xin Yao, and Kazuyuki Murase "Mwmote—Majority Weighted Minority Oversampling Technique for Imbalanced Data Set Learning" IEEE Transactions on Knowledge and Data Engineering, Vol. 26, No. 2, February 2014, Pp 405-426.

[16] Mikel Galar, Alberto Fern´Andez, Edurne Barrenechea, Humberto Bustince, and Francisco Herrera, "A Review on Ensembles for the Class Imbalance Problem: Bagging-, Boosting-, and Hybrid-Based Approaches" IEEE Transactions on Systems, Man, and Cybernetics—Part C: Applications and Reviews, Vol. 42, No. 4, July 2012.

[17] Taghi M. Khoshgoftaar, Jason Van Hulse, And Amri Napolitano "Comparing Boosting And Bagging Techniques With Noisy And Imbalanced Data" IEEE Transactions On Knowledge And Data Engineering, Vol. 26, No. 2, February 2014, Pp 552-568.

[18] Mr.Rushi Longadge, Ms. Snehlata S. Dongre, Dr. Latesh Malik "Class Imbalance Problem in Data Mining: Review" International Journal Of Computer Science and Network (IJCSN) Volume 2, Issue 1, February 2013.

[19] Khobragade, P.K.; Malik, L.G., "Data Generation and Analysis for Digital Forensic Application Using Data Mining," in Communication Systems and Network Technologies

[20] R. Kohavi and J. R. Quinlan, "Decision-tree discovery," in Handbook of Data Mining and Knowledge Discovery, W. Klosgen and J. M. Zytkow, Eds. London, U.K.: Oxford Univ. Press, 2002, ch. 16.1.3, pp. 267–276.

[21] CSNT), 2014 Fourth International Conference on, vol., no., pp.458-462, 7-9 April 2014

[22] S. R. Safavin and D. Landgrebe, "A survey of decision tree classifier methodology," IEEE Trans. Syst., Man, Cybern., vol. 21, no. 3, pp. 660–674, Jul. 1991.

[23] D. Margineantu, Class probability estimation and cost-sensitive classification decisions, In Proc. of 13th European Conference on Machine Learning, Helsinki Finland, pp: 270 - 281, August 2002

[24] http//archive.ics.uci.edu/ml/datasets/Pima+Indians+Diabetes

[25] Aysegul Askan and Serpil Sayin, SVM classification for imbalanced datasets using a multiobjective optimization framework, Annals of Operations Research, Volume 216, Issue 1, pp 191-203, May 2014.

All rights reserved by *www.ijsrd.com*