ISSN 2394-3777 (Print)
ISSN 2394-3785 (Online)
Available online at www.ijartet.com

International Journal of Advanced Research Trends in Engineering and Technology (IJARTET)
Vol. 3, Issue 9, September 2016

An Efficient Mechanism for Classification of Imbalanced Data

Krithika M V[1], Rajeev Bilagi[2], Dr. Prashanth C M[3]
[1]Post Graduate Student, Dept of CS&E, SCE Bangalore, India.
[2]Mr.Rajeev Bilagi, Assoc. Prof, Dept of CS&E, SCE Bangalore, India.
[3]Dr. Prashanth C M, Prof & HOD, Dept of CS&E, SCE Bangalore, India.

Abstract: In many real world applications, there is wide increment in data generation and storage. The classification algorithms are facing a problem in the categorization of highly imbalanced datasets. Classification methods dealt so far centred only on binary class imbalanced problem. All the classification algorithms are biased towards the majority class ignoring most of the significant samples present in the minority class. To resolve this issue, a method called Hybrid Sampling technique is proposed to deal with multi class imbalanced data. The proposed method is an efficient method because it acts by balancing the data distribution of all the classes and imbibes efficient sample selection strategy to undersample the majority class. Experiments are performed using various classifiers and the results of proposed system prove that the classification prediction rate improves when a balanced data having different category of class groupings are considered.

Keywords: Classification, data mining, Imbalance Problems, K means Clustering, Multi Class Imbalanced data, Sampling Techniques, Stratified Sampling

I. Introduction

Most of the real world applications have to identify the occurrence of rare events from very large datasets. Data mining techniques analyse massive amount of data from various sources and resolve issues of various views by summing them up into useful information [1]. Decision making needs good outlined ways for exploration of data or cognition from various areas. Data mining is the prediction of efficacious information from massive datasets. Classification or categorization plays a pivotal role in the application space of data mining. Classification involves assigning a class label to a set of undefined examples.

Classification becomes a serious issue with highly skewed dataset. The classification algorithms proposed so far dealt with imbalanced binary class problem. Evaluating and negotiating the problem of imbalanced data in multiple class domain has been proposed in this study. Class imbalance [2] problem is a predominant issue in the field of data mining and machine learning techniques. All the classification algorithms are biased towards the majority classes, ignoring most of the minority classes that occur very rarely but are found to be the most important.

1.1 Class Imbalanced Data Problem

Class imbalance problem is said to occur when the count on collection of samples in one class (superior class) is not less than half the count of the other class (inferior classes). A class that has largest count on the collection items is related as majority class (superior/negative class) and the one that has comparatively less count on the collection items is related as minority (inferior) class or a positive class. As the superior class has large number of training instances, the classifiers show desirable accuracy rates upon observing such class but the categorization rate drops down when an inferior class is observed. Classification algorithms [3] on imbalanced dataset show poor performances due to the following reasons:

- ✓ The goal of any classification algorithm is to minimize the overall error rates.
- ✓ They assume the class distribution of different class labels as equal.
- ✓ Misclassification error rates of all the classes are considered to be equal [4].
- ✓ Most of the data mining algorithms assume uniform distribution of records among all the classes.

They blindly assume that all the costs associated with every misclassification is same as the ones that are correctly classified. The situation is substantially different in numerous true applications. The vast majority of the ongoing applications contain dataset with skewed distribution [5]. A skewed dataset is the one which has bigger include of accumulation of collection items in one set than the other [6] [7].

1.2 Effects of misclassification

In medical diagnosis application [5], prediction of the occurrence of rare disease is more important than treating the normal diseases that occur very frequently [7]. For example, consider a disastrous malignant disease such a cancer. As this disease occurs very rarely, the number of patients who are tested positive for this disease belong to the minority class label and the ones tested negative are categorized under superior class label. As the classifier is biased towards the superior class (class consisting of the patients who are tested negative), any patient who is tested positive for the cancer disease will also get classified as a cancer free patient. In this case, missing a cancer patient causes more threat than the false positive errors because he/she may even lose her life if proper medication is not given on time. Class imbalance problem is also additionally seen in various domains, for example, misrepresentation recognition in keeping money operations, system interruption detection [8], overseeing chance and foreseeing malfunctioning of specialized type of proficient equipment’s. When much of the above mentioned situations are observed, the classifier shows poor classification rates on the minority class as the classifier is biased towards the superior class.

1.3 Mitigation of misclassification rates

Techniques that may be accustomed for solving the issue of imbalance class [1][6] may be categorized into the following types:

i] Data Level Approach [7]: This approach tries to rebalance the class distribution by employing preprocessing technique. Preprocessing technique involves the application of methods such as oversampling and undersampling.

ii] Algorithm Level Approach [8]: This approach modifies or adopts the existing algorithms over the imbalanced class distribution and achieves a balanced distribution of both the classes by biasing the classifier towards the minority class.

iii] Cost Sensitive approach [9]: Cost sensitive approach takes misclassification error costs into consideration. It does this by associating higher error costs to each of the misclassified example.

1.4 Sampling

Sampling may be defined as the inference or judgment made on some part of the aggregate or totality that is considered. Sampling can be applied over a dataset either to create/add new samples or to remove few samples from the existing dataset. Sampling is a preprocessing technique. Data sampling may be achieved in two different ways [10]:

Undersampling: Random removal of samples from the majority class is the technique employed by this method to achieve a balanced distribution [11]. Training examples from the majority class are eliminated randomly to get a balanced ratio between the classes that are considered.

Oversampling: Random oversampling method acts by replicating the randomly chosen minority class samples to achieve a balanced distribution on both the classes [12].

II. MATERIALS USED

2.1 Methods Employed and Their Shortcomings

2.1.1 Parallel Selective Sampling Method:

Parallel selective sampling method [13] considered huge amount of imbalanced data and provided solutions for classifying them. Performances were assessed using Parallel Selective Sampling (PSS), a method that reduces the imbalance in large data sets by selecting the data from the superior class.

Disadvantage: As PSS is an undersampling method, it removes or eliminates the instances from superior class. These randomly removed samples affect the class distribution because, the eliminated samples may be the significant samples that are considered to be important during classification. Eliminating such significant samples may degrade the classifiers performance.

2.1.2 Neighborhood Based Rough Set Boundary Synthetic Minority Oversampling Technique:

Hu, F., Li, H. [14] proposed an oversampling method, called Neighborhood Based Rough Set Boundary Synthetic Minority Oversampling Technique (NRS Boundary SMOTE), to achieve a balanced distribution. The minority class samples present in the boundary region are considered for oversampling.

Disadvantage: Filtering the synthetic samples take more time and hence there is a difficulty in processing the large datasets. Also, oversampling method consists of the instances or the datasets that do not represent the universal sample.

2.1.3 Borderline SMOTE Technique:

Borderline SMOTE [15]: is a oversampling technique (SMOTE) that addresses the issue relating imbalanced classification of data sets. They presented two new oversampling techniques based on SMOTE namely, borderline SMOTE1 and borderline SMOTE2.

Disadvantage: Borderline SMOTE suffer from curse of dimensionality because they rely heavily on Euclidean distance. They focused only on two class imbalance problem.

2.2 Existing System

- ✓ Undersampling method incurs the problem of loss of valuable information.
- ✓ Random oversampling method may not represent the underlying domain.
- ✓ The traditional methods do not address the case where more than one minority classes are the class of interest.

2.3 Problem Statement

"To develop a methodology which balances the multi class imbalance data using hybrid sampling technique, that uses a combination of both oversampling and undersampling methods with efficient sample selection".

2.4 Scope of the Project

- ✓ To increase the accuracy of the classifier by transforming a multi class imbalanced data to a balanced data.
- ✓ To annihilate the issue of class imbalance problem by eliminating insignificant samples that exists in majority class instead of random undersampling.

2.5 Proposed System

The proposed method tries to employ an efficient sample selection strategy that involves selecting samples from the superior class that has significant importance instead of random undersampling. This helps in maintaining the class distribution of original training set, produces reliable results and thus enhances the classification accuracy of the classifier.

III. Methodology

Measure the Class distribution of the given imbalanced data. Compute the mean of all the classes considered and select it as reference point to sample the data accordingly. In order to match the mean and obtain a balanced distribution, the minority class records are oversampled and superior class records are undersampled.

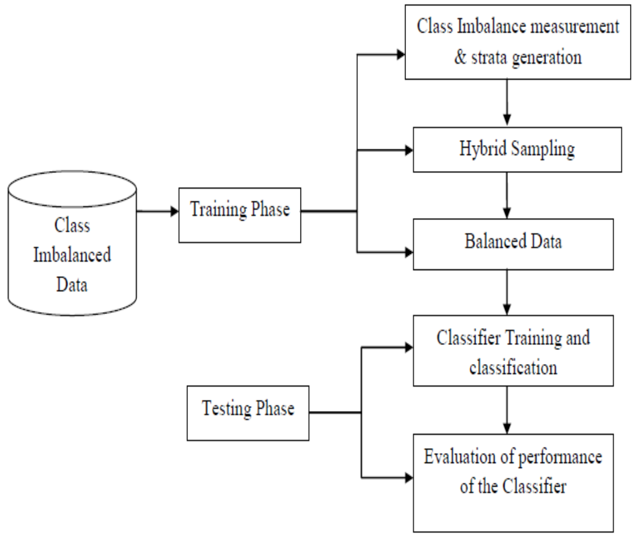
Class Imbalanced Data
Training Phase
Class Imbalance measurement & strata generation
Hybrid Sampling
Balanced Data
Testing Phase
Classifier Training and classification
Evaluation of performance of the Classifier

Figure 3.1: Proposed system architecture

Figure 3.1 summarizes the proposed technique for handling multi class imbalance problem

Advantages:

1. It is applicable to the cases where one or more than one minority class is of interest.

2. Mean is used as a reference point to sample all the class records without any cross check for balancing and multiple iterations. It reduces the processing time as all the records are balanced in a single stage of processing

3. There is increase in the accuracy of classification algorithm as the effect of demerits of oversampling and undersampling techniques are balanced.

3.2 Modules

The methodology involved in balancing the imbalanced class distribution can be divided into 3 phases.

1. Class Imbalance measure
2. Strata generation
3. Hybrid Sampling

3.2.1 Class Imbalance Measure: Given a dataset, measure the number of records distributed for each class as depicted in table 3.1.

Table 3.1 Class distribution

Class	No.of Records
Class #1	17

Class #2	37
Class #3	666
Mean	240

Pseudo code for Class Imbalance Measure

```
Input: Class Imbalanced Dataset
Output: Class Distribution
Scanner = File.Open("Dataset File Path)
Map<Integer,Integer> classCounter = 0
WHILE(Scanner.hasNextDataPoint)
START
    dataPoint = Scanner.NextDataPoint()
    ClassIndex = dataPoint.getClassIndex()
    Count=classCounter.get(ClassIndex).
getPreviousCount( )
    Count = Count + 1
   classCounter.replace(ClassIndex, Count)
END
Mean = 0.0
Sum = 0.0
FOR each classIndex in classCounter
Sum = Sum + classCounter.getCount(classIndex)
END FOR
Mean = Sum / No.ofClasses
FOR each classIndex in classCounter
    Print(classCounter.get(classIndex).getCount( ))
END FOR
```

3.2.2 Strata Generation: The subpopulation of individual class records (table 3.2) separated for sample selection may be referred to as strata. Number of strata is equivalent to number of class present in a data file. Each strata consists of list of records present in each of the class.

Table 3.2 Strata generation

Class	No.of Records	Strata #
Class #1	17	1
Class #2	37	2
Class #3	666	3

Pseudo code for Stratification of Dataset

```
Input: Class Imbalanced Dataset
Output: Stratified Data
Scanner = File.Open("Dataset File Path)
Map<Integer,List<DataPoint>> classStratas
WHILE(Scanner.hasNextDataPoint)
START
    dataPoint = Scanner.NextDataPoint()
    ClassIndex = dataPoint.getClassIndex()
    dataPointList = classStrata.get(ClassIndex)
    dataPointList.add(dataPoint)
END
```

3.2.3 Hybrid Sampling

a. Simple Random Sampling with Replace (Oversampling): The strata's having records less than the mean value of a given dataset are selected and are sampled randomly in this module. A record which is selected once is again eligible for the process of resampling. This condition is called "sampling with replacement".

b. Stratified Random Sampling without Replace (Undersampling): This module clusters the data points from the superior class strata and picks the records randomly from different clusters, proportional to the cluster size.

E.g. For the above example in table 3.1, Mean value is 240 and class #3 is the majority class containing 666 records. Assume that the data points of this class are represented in 3 clusters of different sizes as given below.

Cluster 1: 48 Data points
Cluster 2: 280 Data points
Cluster 3: 338 Data points

The data points of the majority class is distributed in 1:5:7 ratio in the clusters.

Required data points are 240.
Ratio Total = 1+5=7 = 13.

Minimum Records to be fetched from a cluster = 240/13 = 18

Now fetch,

18 records from Cluster 1
90 records from Cluster 2

Remaining 240 – (18+90) = 240 – 108 = 132 records from cluster 3 (Larger Cluster)

Pseudo code for Hybrid Sampling Technique

a. Oversampling

```
Input: Class Stratified Data having data points count less
than Mean, Mean Value
Output: Oversampled Data
```

```
    List<dataPoints> OverSampledList
    SampleCount = DataPointSize
    WHILE SampleCount < Mean
    START
      dataPointID = RandomNumberGenerator(0 to DataPointSize)
      dataPoint = DataPointsList.get(dataPointID)
      OverSampledList.add(dataPoint)
      SampleCount = SampleCount + 1
    END
```

b. Undersampling

```
Input: Class Stratified Data having data points count More than Mean, Mean Value
Output: Undersampled Data
---
List<dataPoints> UnderSampledList
ClusterList=KMeansClustering(Original DataPoint List , 4)
Map<ClusterID, Count> clusterSize
FOR each cluster in ClusterList
START
ClusterSize.put (ClusterID, cluster.count( ) )
END
Ratio = CalculateRatio(ClusterList)
MinimumDataPoints = Mean / Sum(Ratios)
FOR each cluster in ClusterList
START
No.ofDataPointsToFetch = MinimumDataPoints * Ratio[x]
UnderSampledList.add(Cluster[x].getRandomDataPoints(No.ofDataPointsToFetch)
END
```

Table 3.3 Application of Hybrid Sampling Technique

Class	No.of Records	No.of Records Inserted/Deleted	Sampling Technique used	# of Records after sampling
Class #1	17	+223	Oversampling	240
Class #2	37	+203	Oversampling	240
Class #3	666	-426	Undersampling	240

IV. Results And Analysis

For experimental set up, imbalanced thyroid dataset consisting of multiple class is extracted from UCI repository and is implemented using Net beans 8.0.2 IDE. Thyroid dataset consists of 720 instances, 21 attributes and 3 different class labels. Classifier is trained with a balanced dataset and classification is performed using different classifiers such as KNN, Decision tree, Rule based classifier, Random Forest and Simple Logistic Regression.

This section provides experimental and comparison results of various classifiers using different evaluation parameters such as precision, recall, FP rate and f measure with respect to balanced and imbalanced dataset in a multi class imbalanced domain.

Sensitivity (true positive rate): $$\frac{TP}{TP+FN} \quad (1)$$

Specificity(true negative rate): $$\frac{TN}{TN+FP} \quad (2)$$

False positive rate : $$\frac{FP}{FP+TN} \quad (3)$$

Precision : $$\frac{TP}{TP+FP} \quad (4)$$

F-Measure : $$\frac{2.\ Precision.\ Recall}{Precision+\ Recall} \quad (5)$$

The experimental results of imbalanced data classification (performed using various classifiers) produce very low classification rates resulting in degradation of performance of the classifier. Classification rates on the minority class instances are very low in imbalanced dataset. Hence the proposed method is used to balance the dataset thereby allowing the classifier to evaluate the model. Comparison results performed with regard to imbalanced data and balanced data show that there is increased level of performance of the classifier for a balanced dataset using the proposed system. Figure 4.1, 4.2 and 4.3 shows the comparison rates of precision, recall and f measure values for imbalanced and balanced dataset respectively. The graph is plotted between different classifiers in the X axis and their respective precision, recall and f measure rates in Y axis.

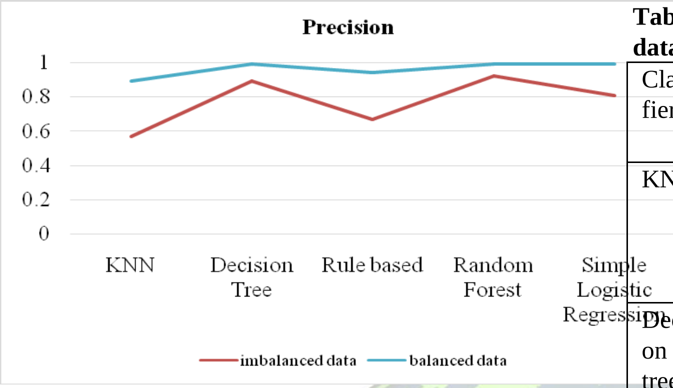

Figure 4.1: Precision rate for imbalance and balanced dataset

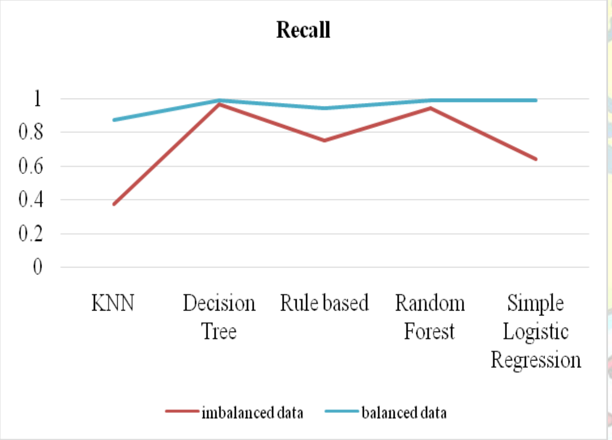

Figure 4.2: Recall rate for imbalance and balanced dataset

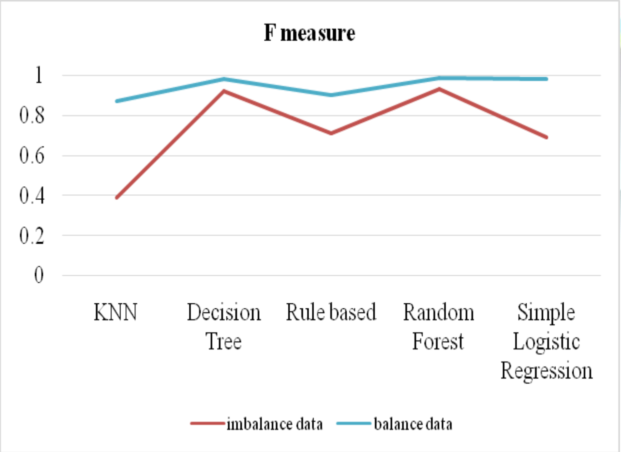

Figure 4.3: F measure rate for imbalance and balanced dataset

Table 4.1 Classification rates of imbalanced and balanced dataset on various classifiers.

Classifier	Dataset	TPR	FPR	Precision	Recall	F measure
KNN	Imbalanced	0.373	0.3	0.57	0.373	0.39
	Balanced	0.87	0.06	0.89	0.87	0.87
Decision tree	Imbalanced	0.963	0.0103	0.896	0.963	0.926
	Balanced	0.99	0.006	0.99	0.99	0.986
Rule based classifier	Imbalanced	0.75	0.051	0.67	0.75	0.71
	Balanced	0.94	0.025	0.94	0.94	0.93
Random Forest	Imbalanced	0.946	0.027	0.92	0.946	0.936
	Balanced	0.99	0.004	0.99	0.99	0.986
Simple Logistic Regression	Imbalanced	0.643	0.19	0.81	0.643	0.69
	Balanced	0.99	0.004	0.99	0.99	0.986

Table 4.1 shows the effects of classification when classifying an imbalanced dataset and the effects of classification after balancing the imbalanced data using the proposed hybrid sampling technique.

V. Conclusion

The classification of imbalance data that exists in real world produces very low classification rates resulting in degradation of performance of the classifier. To overcome this problem, a data pre-processing technique called Hybrid Sampling technique is proposed to generate balanced data from multi class imbalanced data. The proposed method uses an efficient sample selection strategy to pick the samples from the majority class in order to yield reliable results. All the class records are balanced in a single stage of processing resulting in reduced processing time. This method is used to balance the dataset thereby allowing the classifier to evaluate the model. The classifiers performance improves quite effectively after balancing the imbalanced data using the efficient hybrid sampling technique as shown in table 4.1 proposed method is evaluated using various classifiers. The accuracy performance of simple logistic regression and random forest ensemble is 99.3% which outperforms all the other classifiers. The classification rate of decision tree is about 98.88% which is more than rule based classifier (94.3%) and KNN classifier (86.8%).

ACKNOWLEDGMENT

I express my deep thanks to Dr. Prashanth C M, Professor & Head, Department of CS&E for warm hospitality and affection towards me. I thank the anonymous referees for their reviews that significantly improved the presentation of this paper. I am thankful to Mr. Rajeev Bilagi for his variable advice and support extended to me without whom I could not complete my paper. Words cannot express my gratitude for all those people who helped me directly or indirectly in my endeavor. I take this opportunity to express my sincere thanks to all the staff members of CS&E department of SCE for their valuable suggestion.

REFERENCES

[1]. Reshma C.Bhagat, R.C., Sachin S. Patil "Enhanced SMOTE Algorithm for Classification of Imbalanced Big-Data using Random Forest". Advanced Computer Conference (IACC), 2015, IEEE international, pp 403 - 408.

[2]. Shaza M.Abd Elrahman and Ajith Abraham "A Review of Class Imbalance Problem". Journal of Network and Innovative Computing, Volume 1(2013) pp.332-340.

[3]. Vaishali Ganganwar. "An overview of classification algorithms for imbalanced datasets" International Journal of Emerging Technology Vol.2, 4(2012).

[4]. C.V. KrishnaVeni,T. Sobha Rani "On the Classification of Imbalanced Datasets", IJCST, Vol . 2, SP 1, December 2011.

[5]. Mikel Galar,Fransico, "A review on Ensembles for the class Imbalance Problem: Bagging, Boosting and Hybrid- Based Approaches" IEEE Transactions On Systems, Man, And Cybernetics—Part C: Application And Reviews, Vol.42,No.4 July 2012.

[6]. Chawla, N. V., Bowyer, K. W., Hall, L. O., & Kegelmeyer, W. P. (2002). "SMOTE: synthetic minority over-sampling technique". Journal of artificial intelligence research, 16(1), 321-357.

[7]. Chawla, N. V., Lazarevic, A., Hall, L. O., & Bowyer, K. W., (2003). "SMOTE Boost: Improving prediction of the minority class in boosting". In Knowldge Discovery in Database: PKDD 2003 (pp. 107-119). Springer Belin Heidelberg.

[8]. Park, B. J., Oh, S. K., & Pedrycz, W. (2013). "The design of polynomial function based neural network predicators for detection of software defects". Information Sciences, 229, 40-57.

[9]. Takshak Desai., Udit Deshmukh, Prof. Kiran Bhowmick "Machine Learning for Classification of Imbalanced Big Data" International Journal on Recent and Innovation Trends in Computing and Communication(IJRITCC), October 2015, pp.6049 - 6053.

[10]. Peng Liu, Lijun Cai, Yong Wang, Longbo Zhang "Classifying Skewed Data Streams Based on Reusing Data" International Conference on Computer Application and System Modeling (ICCASM 2010).

[11]. Xinjian Guo, Yilong Yin1, Cailing Dong, Gongping Yang, Guangtong Zhou,"On the Class Imbalance Problem" Fourth International Conference on Natural Computation, 2008.

[12]. Alexander Yun-chung Liu, B.S. "The Effect of Oversampling and Undersampling on Classifying Imbalanced Text Datasets" August 2004.

[13]. Annarita D'Addabbo, Rosalia Maglietta. "Parallel selective sampling method for imbalanced and large data classification". Institute of Intelligent Systems for Automation - National Research Council, Volume 62. 5(2015)., pp 61-67.

[14]. Hu, F., Li, H. (2013). "A novel boundary oversampling algorithm based on neighboured rough set model: NRS Boundary SMOTE". Mathematical Problems in Engineering, 2013.

[15]. Hui Han, Wen Yuan Wang, Bing Huan Mao. "Borderline SMOTE: A New OverSampling Method in Imbalanced Data Sets Learning" Springer Berlin Heidelberg., Vol. 3644, pp 878-887 (2005)