ISSN No:-2456 –2165

Handling Imbalance Class Problem Using Ensemble Classifier

Snehlata S. Dongre,
Research Scholar
Department of CSE. GHRCE, Nagpur, India

Dr. L. G. Malik,
Former Professor
Department of CSE. GHRCE, Nagpur, India

Achamma Thomas,
Assistant Professor
Department of CSE. GHRCE, Nagpur, India

Abstract—Rare class problem become very popular as many researchers focused to this area. Many applications generate imbalanced datasets in real life. In imbalanced data the ratio of various class samples are not balanced. Classification becomes difficult because of this imbalanced nature of data. Handling rare class problem is an issue in the Data mining. More number of samples belonged to the class is termed as Majority class and less number of sample belonged to the class named as minority class. Sometimes the classification is biased towards majority class samples and ignoring the minority class samples. Because of this the overall accuracy may be good but the class wise accuracy is poor. Various techniques for handling rare class problem have discussed. In this paper, an algorithm Ensemble Boosting Classifier has been proposed for handling rare class problem. Algorithm has been tested for real imbalance datasets and results are good.

Keywords— Class Imbalanced Problem, Skewed Data, Rare Class Problem, Data Mining

I. INTRODUCTION

Rare class problem is the one of the main issue in the data classification. Skewed data problem, Class imbalanced problem and rare class problems all are same terms and interchangeably used in this paper. Medical data [1] is the one of the example of imbalanced dataset as the samples for desease class are much less than the samples for normal class. Here the minority class is 'desease' and majority class is 'normal'. It is most important to classify the minority class as well as majority class accurately or it will lead to wrong diagnostics. Minority class samples are less but plays very important role in some areas. The other examples of imbalanced datasets are intrusion detection [2], fault detection [3], anomaly detection [4], detection of fraudulent telephone calls etc. The overall accuracy of classification method is good as it has been well trained with majority class samples but the class detection accuracy is poor for minority class samples as they are insufficient for the training. There is a need for efficient algorithm that can handle such kinds of imbalanced datasets.

This paper is the extended version of our previous papers [22][23]. Fig. 1 shows rare class problem. Where star shows the minority class samples, and triangle shows the majority class samples.

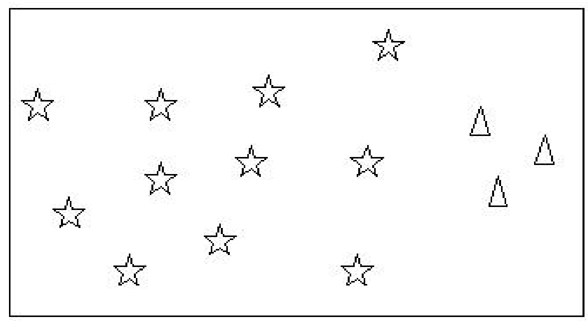

Fig. 1 Rare Class Problem in Dataset

This paper organized as follows: Section 2 presents the related wok, which discusses the different approaches used for classification of the rare class problem in the literature. Section 3 explains the proposed Algorithm. Experimental setup is discussed in section 4. Results and discussions are shown in the section 5 and, Section 6 concludes this paper.

II. RELATED WORK

Many approaches are mentioned in the literature to address the rare class problem. Most of the authors have classified these approaches in mainly two categories [5]- 1) Sampling based solution and 2) Algorithmic based solution. Few authors also consider the 3) Feature selection as the solution for rare class problem. But we believe that there are three main categories of approaches to handle the rare class problem-1) Data level approach- here the main concentration is to balance the data set and after that classify the data. Normal classifier even gives the good results as the dataset is balanced. This mainly uses two techniques- Under sampling [5][6] majority samples and Oversampling [5][6] minority samples. In under sampling, majority class samples are removed to balance dataset whereas oversampling adds few samples to balance it. Each has its pros and cons. Best is to use the combination of oversampling & under sampling. 2) Algorithmic approaches, in which algorithms are designed such that they can handle rare class

ISSN No:-2456 –2165

problem, this category includes single classifier and ensemble classifier and 3) Hybrid Approach, which is the combination of data level approach and algorithmic approach. Most of the strategies fall under this category as many strategies uses data sampling approach for balancing the data as well as classifier that have the capability to handle rare class problem. So this is combination of data sampling and classifier both.

A. Data level Approach

Synthetic minority oversampling technique (SMOTE) [7] is a very popular approach that synthetically generates the minority samples to balance dataset. It uses the oversampling technique. Modified synthetic minority oversampling technique (MSMOTE) [8] is the improvement in the SMOTE. It is quite similar as SMOTE the difference is in selection of Nearest Neighbor Synthetic minority oversampling technique (SMOTE) [7] is a very popular approach that synthetically generates the minority samples to balance dataset. It uses the oversampling technique. Modified synthetic minority oversampling technique (MSMOTE) [8] is the improvement in the SMOTE. It is quite similar as SMOTE the difference is in selection of Nearest Neighbor for the generation of synthetic minority samples.

B. Algorithmic Approach

Xiaowan Zhang et al. [9] have proposed Cost Free Learning strategy for handling the class imbalanced problem. They said that there are mainly two categories Cost Free Learning (CFL) and Cost Sensitive Learning (CSL). Mutual information is used here. But this strategy adds additional computational cost to different approaches. He et. al [10] uses the SVM for handling class imbalanced problem. Authors have suggested two modifications in SVM so that it can deal with data imbalanced problem. Yubin Park et al [11] have proposed ensemble of decision trees for handling class imbalance problem. Properties α-divergence is used for this purpose. Peng Wang et al [12] have discussed the concept of granularity for classifier and by using this concept they have proposed a low granularity classifier that can deal with concept drift as well as class imbalance problem.

C. Hybrid Approach

Balanced Boost Technique is a Hybrid approach that uses data level approach along with the algorithmic approach proposed by H. Wei et al. [13]. It uses feature selection method named weighted symmetrical uncertainty with ensemble algorithm Balanced Boost to deal with the rare class problem. This approach gives good results for network traffic data. S. Wang et al. [14][15] have proposed two approaches Over sampling based Online Bagging(OOB) and Under sampling based Online Bagging(UOB). Both are based on bagging ensemble classifier approach. Performed good for imbalanced dataset but these have not bothered about the imbalanced ratio even the resembling rate is not at all correlated with imbalanced rate. OOB and UOB have been analyzed by the authors and improved to overcome its disadvantages. C. Seiffert et al. [16] have proposed hybrid approach named as RUS Boost of data sampling with boosting which is a combination of data level approach and algorithmic approach. RUS Boost uses the Random under sampling Technique with Boosting technique for handling data imbalanced problem. Random under sampling deletes the instances randomly from the dataset until it balance the dataset. RUS Boost is reducing training time also increasing the performance. But it is using under sampling technique, so it deletes few important data from the dataset that may play vital role in the data classification.

III. PROPOSED WORK

A. Ensemble Classifier

Ensemble classifier is a group of classifiers with an aim to produce the better results than a single classifier. The basic idea in this is to train each individual classifier also known as base classifier or base learner. After the training, they start prediction. To get the final global prediction the local prediction of each base learner is considered. For the computation of the global prediction the ensemble classifier generally used two methods: Bagging and Boosting[17][18]. The main focus of this thesis is on Boosting. Whenever the concept of stream data comes, ensemble classifier is much better than the single classifier. Ensemble classifier has the capability to cope with the concept drift by updating the model frequently. Ensemble classifier tries to deal with concept drift and gives the promising accuracy. But along with the accuracy, there were two more important parameters time and memory. So to get the good performance, selection of base learner, the methodology that combines the local predictions to get global prediction is very important.

B. Weighted Majority Algorithm

The main objective of the ensemble is to represent the current underlying concept and thus be able to classify the incoming records with high accuracy. One of the most important challenges of stream ensemble approaches is on how to weight the models in the ensemble. Weighted Majority approach has been used here. The Weighted Majority approach maintains as its concept description an ensemble of learning algorithms, each referred to as an expert and each with an associated weight. Given an instance, the performance element polls the experts, each returning a prediction for the instance. Using these predictions and expert weights, the algorithm returns as the global prediction the class label with the highest accumulated weight. The learning element, given a new training example, first polls each expert in the manner described previously. If an expert predicts incorrectly, then its weight is reduced by the multiplicative constant β. The algorithm then determines the global prediction. If it is incorrect, then the algorithm creates a new expert with a weight of one. The algorithm normalizes expert weights by uniformly scaling them such that the highest weight will be equal to one. This prevents any newly added experts from dominating the decision making of existing ones. The algorithm also removes experts with weights less than the user-defined threshold θ. Finally, algorithm passes the training example to each expert's learning element. Note that normalizing weights and incrementally training all experts gives the base learners an opportunity to recover from concept drift. Large and noisy problems required the parameter p, which governs the frequency that the Weighted Majority creates experts, removes them, and updates their weights.

C. Adaptive Sliding Window [14]

In data streams environment data comes infinitely and huge in amount. So it is impossible to stores and processes such data fast. To overcome these problems window technique proposed. Window strategies have been used in conjunction with mining algorithms such as, externally to the learning algorithm; the window system is used to monitor the error rate of the current model, which under stable distributions should keep decreasing or at most stabilize; when instead this rate grows significantly, change is declared and the base learning algorithm is invoked to revise or rebuild the model with fresh data. A window is maintained that keeps the most recent examples and according to some set of rules from window, older examples are dropped. Windowing technique does not store whole window explicitly but instead of this, it only stores statistics required for further computation.

D. Hoeffding Tree

The Hefting tree induction algorithm induces a decision tree from a data stream incrementally, briefly inspecting each example in the stream only once, without a need for storing examples after they have been used to update the tree. The only information needed in memory is the tree itself, which stores sufficient information in its leaves in order to grow, and can be employed to form predictions at any point in time between processing training examples.

Algorithm: Ensemble Boosting Classifier

$h_1, \ldots\ldots, h_m$ Experts

Input: d, a set of class labeled

Base learning algorithm is Hoeffding Tree

Output: a composite model

1. Set the model Em for m Є {1, 2, 3,……,M}
2. Set the value of β, ø, and p
3. Initialize w_1, w_2, $w_3 \ldots$ w_m weight of each classifier where (m Є {1, 2, 3,……,M})
4. Initialize the window W
5. Initialize the width, variance, and total
6. For each t>0
7. Setinput (X_t, W)
8. The window W is partitioned into two sub-windows
9. If change detected & dropping the old window
10. Update width, variance and total
11. Change Alarm The current data of new window is used to update the learning model end for
12. Each expert gives prediction for the data
13. Weights of experts are updated
14. If the expert predicts incorrectly then the weight of this expert is reduced by multiplicative constant β
15. Algorithm determines the global prediction
16. If it is incorrect then algorithm creates a new expert with weight 1
17. Normalize the weights of experts
18. Algorithm will remove the expert with weight less than ø
19. If the data is large and noisy set the p, frequency that the algorithm creates expert, removes them and update their weights.

The Pseudo code of Ensemble Boosting Classifier(EBC) is shown in Algorithm. In this model, all the base model of ensemble is set, how many classifiers you want to group, which type of classifier is taken as the base learner. Then in step 2 the different parameters β, ø and p are set. In step 3 the weights are assigned to each classifier initially, and then afterward they are updated by the model. Window W is initialized and its width, variance, and total have also been initialized in step 4 & 5. If the data is available that data is kept in the window W. The Window W is divided into two sub-windows of large enough size. If the change is detected between these two windows then we are keeping the new window and dropping the old window. Updating the width, variance and total accordingly and raise a change alarm to show that the change has been detected. This new window is now used to update the model. The above procedure has shown in the steps from 6-13. In step 14-20, once the data is given to the model, each expert gives the local prediction. As per their predictions are weights are updated. If the prediction of an expert is incorrect, then its weight decreases by the multiplicative constant β. In next step, the algorithm calculates the global prediction as the class label with the highest accumulated weight. If the global prediction is incorrect it means we need to update the model by adding a new expert in the ensemble with weight 1. The weights of all experts are normalized to eliminate the domination of newly added expert to others. An expert whose weight is below the ø threshold is removed from the ensemble. In the last step 21 if the data is large and noisy set p, the frequency that the algorithm creates experts, removes them and update their weights.

IV. EXPERIMENTAL SETUP

The experiments were performed on a 2.20 GHz Intel Core 2 Duo processor with 2 GB RAM, running on UBUNTU 11.04. The MOA (an experimental tool for Massive Online Analysis) [20] framework has been used.

A. Datasets used

To validate the efficiency and effectiveness of our proposed algorithms, we have performed extensive experiments on benchmark real datasets

a). Car Dataset

Car Evaluation Database. Number of Instances: 1728. Number of Attributes: 6. This dataset is an imbalanced dataset.

Class Distribution (number of instances per class)
class N N[%]
----
unacc 1210 (70.023 %)
acc 384 (22.222 %)
good 69 (3.993 %)
v-good 65 (3.762 %)

b). Nursery Dataset

Nursery dataset having Number of Instances: 12960. Number of Attributes: 8. This dataset is an imbalanced dataset.

Class Distribution (number of instances per class)
class N N[%]
----
not_recom 4320 (33.333 %)
recommend 2 (0.015 %)
very_recom 328 (2.531 %)
priority 4266 (32.917 %)
spec_prior 4044 (31.204 %)

c). Connect-4 Dataset

This database contains all legal 8-ply positions in the game of connect-4 in which neither player has won yet, and in which the next move is not forced. Number of Instances: 67557.Number of Attributes: 42, each corresponding to one connect-4 square. This dataset is an imbalanced dataset.
Class Distribution:
44473 win (65.83%),
16635 loss (24.62%),
6449 draw (9.55%).

V. RESULTS AND DISCUSSION

Proposed model EBC is tested for the rare class problem for real data Real-time imbalance datasets are collected from the UCI Machine learning repository Real Datasets. We have used Car dataset, Nursery dataset and Connect-4 dataset as these are imbalanced datasets where the numbers of one class examples are more than the number of other class examples. Comparison of experimental results with the well known algorithms like OCBoost [21], OzaBag [17][18], OzaBagADWIN [19], OzaBoost [17][18] data stream algorithms has been done.

A. Results using Car Dataset

Algorithm	Class name	Actual	Detected	Detection rate	Overall Accuracy
OCBoost	**unacc**	1209	988	81.72	70.75
	acc	384	234	60.94	
	vgood	65	0	0.00	
	good	69	0	0.00	
OzaBag	**unacc**	1209	1209	**100.00**	70
	acc	384	0	0.00	
	vgood	65	0	0.00	
	good	69	0	0.00	
OzaBagADWIN	**unacc**	1209	1209	**100.00**	70
	acc	384	0	0.00	
	vgood	65	0	0.00	
	good	69	0	0.00	
OzaBoost	**unacc**	1209	1119	92.56	76.72
	acc	384	206	**53.65**	
	vgood	65	0	0.00	
	good	69	0	0.00	
EBC	**unacc**	1209	1206	99.75	**82.62**
	acc	384	188	48.96	
	vgood	65	14	**21.54**	
	good	69	19	**27.54**	

Table 1: Detection Rate and Overall Accuracy for Car Dataset Using EBC Algorithm

Table I shows the results obtained for Car dataset using EBC. The detection rate of each class is calculated to show the class wise classification. The overall the accuracy is also calculated. It has been observed that the EBC is having better accuracy for both the classes good and good which are having very fewer numbers of examples as compared to other classes. Even the overall accuracy is also better than the other compared algorithms.

B. Results using Nursery Dataset

Algorithm	Class name	Actual	Detected	Detection rate	Overall Accuracy
OCBoost	Recommend	1	0	0.00	54.22
	Priority	4266	2962	69.43	
	Not_recc	4320	4260	98.61	
	Very_recc	328	0	0.00	
	Spec_prior	4404	0	0.00	
OzaBag	Recommend	1	0	0.00	79.96
	Priority	4266	4051	**94.96**	
	Not_recc	4320	4258	98.56	
	Very_recc	328	0	0.00	
	Spec_prior	4404	2341	53.16	
OzaBag ADWIN	Recommend	1	0	0.00	81.16
	Priority	4266	4033	94.54	
	Not_recc	4320	4258	98.56	
	Very_recc	328	0	0.00	
	Spec_prior	4404	2520	57.22	
OzaBoost	Recommend	1	0	0.00	86.49
	Priority	4266	3754	88.00	
	Not_recc	4320	4258	98.56	
	Very_recc	328	12	3.66	
	Spec_prior	4404	3496	79.38	
EBC	Recommend	1	0	0.00	**88.54**
	Priority	4266	4040	94.70	
	Not_recc	4320	4318	**99.95**	
	Very_recc	328	265	**80.79**	
	Spec_prior	4404	3170	**71.98**	

Table 2: Detection Rate and Overall Accuracy for Nursery Dataset Using EBC Algorithm

Table 2 shows the results obtained for Nursery dataset using EBC. The detection rate of each class is calculated to show the class wise classification. The overall the accuracy is also calculated. It has been observed that the EBC is having better accuracy for three of the classes Not_recc, very_recc and spec prior but it has failed to detect the recommended class. Ever the other classifiers could not able to detect that class accurately as the total number of examples is 1 for the recommended class which is very much less than the numbers of examples as compared to other classes, so having a high probability of misclassification. The overall accuracy is also better than the other compared algorithms.

C. Results using Connect-4 Dataset

Algorithm	Class name	Actual	Detected	Detection rate	Overall Accuracy
OCBoost	win	44472	40210	90.42	61.79
	draw	6449	1535	**23.80**	
	loss	16635	0	0.00	

OzaBag	**win**	44472	43102	**96.92**	68.23
	draw	6449	0	0.00	
	loss	16635	2993	17.99	
OzaBagADWIN	**win**	44472	41848	94.10	67.55
	draw	6449	0	0.00	
	loss	16635	3788	22.77	
OzaBoost	**win**	44472	40315	90.65	67.8
	draw	6449	2	0.03	
	loss	16635	5491	33.01	
EBC	**win**	44472	39890	89.70	**74.4**
	draw	6449	1008	15.63	
	loss	16635	9364	**56.29**	

Table 3: Detection Rate and Overall Accuracy for Connect-4 Dataset Using EBC Algorithm

Table 3 shows the results obtained for Connect-4 dataset using EBC. The detection rate of each class is calculated to show the class wise classification. The overall the accuracy is also calculated. It has been observed that the EBC is having better accuracy for the class loss. Even the overall accuracy is also better than the other compared algorithms.

D. Confusion Matrix Has Been Calculated for Nursery

	Recommend	**Priority**	**Not_recc**	**Very_recc**	**Spec_prior**
Recommend	0	0	9	159	58
Priority	2	4040	0	0	0
Not_recc	1	0	4318	0	0
Very_recc	53	0	10	265	0
Spec_prior	874	0	0	0	3170

Confusion matrix has been calculated for the Nursery dataset to show the detail of classification parameters more clearly.

EBC is able to handle the rare class problem and giving good detection rate & overall accuracy as compared to other mention algorithms in literature. Without applying any special sampling strategy for rare class even it is working properly and results are acceptable.

VI. CONCLUSIONS

Class Imbalanced problem is an issue in the real world applications. Ignoring the minority class samples leads to high risk in real life scenario. Various approaches for handling the rare class problem has been discussed. Ensemble Boosting classifier (EBC) has been tested for the rare class problem and it shows better results as compared to the other approaches. EBC gives good class detection rate as well as the overall accuracy is also good. EBC is an algorithmic approach in which we have not used any data level approach still it is working properly and the results are acceptable. The future work can be applying the Hybrid approach for handling the rare class problem which uses algorithmic approach and data level approach both to increase the performance.

REFERENCES

[1]. M. Mazurowski, P. Habas, J. Zurada, J Lo, J. Baker and G. Tourassiet, “Training neural network classifiers for medical decision making: The effects of imbalanced datasets on classification performance,” Neural networks Vol. 21, No. 2, pp 427-436, 2008.

[2]. S. Dongre, K. Wankhade, “Intrusion Detection System Using New Ensemble Boosting Approach,” International Journal of Modeling and Optimization, Vol. 2, No. 4, pp 488-492. August 2012.

[3]. Z. yang, W. Tang, A. shintemirov, and Q. wu, “Association rule miningbased dissolved gas analysis for

fault diagnosis of power transformers," IEEE Transaction of Sytem, Man, Cybernatics. C, Appl. Rev., vol. 39, no. 6, pp. 597-610, 2009.

[4]. W. Khreich, E. Granger, A. Miri, R. Sabourin, "Iterative Boolean combination of classifiers in the ROC space: An application to anomaly detection with HMMs," Pattern Recognition Vol. 43, No. 8, pp 2732-2752, 2010.

[5]. R. Longadge, S. Dongre, L. Malik, "Class Imbalance Problem in Data Mining: Review," International Journal of Computer Science and Network (IJCSN) Volume 2, No 1, February 2013.

[6]. T. Lakshi, Ch. Prasad, "A study on classifying Imbalanced Datasets," Proc. International Conference on Networks & Soft Computing. 2014.

[7]. N. Chawla, K. Bowyer, L. Hall, and W. Kegelmeyer, "SMOTE: synthetic minority over-sampling technique," Journal of Artificial Intelligence Research, vol. 16, pp. 321–357, 2002.

[8]. S. Hu, Y. Liang, L. Ma, and Y. He, "MSMOTE: Improving classification performance when training data is imbalanced," Proc. International Workshop Computer Science and Engineering, vol. 2, pp. 13–17, 2009.

[9]. X. Zhang and B. Hu, "A New Strategy of Cost-Free learning in the Class Imbalance Problem'" IEEE Transactions on Knowledge and Data Engineering. Vol. 25, No. 12, pp 2872-2885, December 2014.

[10]. H. He and A. Ghodsi, "Rare class Classification by Support Vector Machine," Proc. International Conference on Pattern Recognition IEEE, pp 548-551, 2010.

[11]. Y. Park and J. Ghosh, "Ensembles of α-Trees for Imbalanced Classification Problems" IEEE Transactions on Knowledge and Data Engineering, Vol. 26, No. 1, pp 131-143, 2014.

[12]. P. Wang, H. Wang, X. Wu, W. Wang, and B. Shi, "A Low-Granularity Classifier for Data Streams with Concept Drifts and Biased Class Distribution," IEEE Transactions on Knowledge and Data Engineering, Vol. 19, No. 9, pp 1202-1213, 2007.

[13]. H. Wei, B. Sun and M. Jing, "BlancedBoost: A Hybrid Approach for Real-time Network Traffic Classification," Proc. IEEE International Conference on Computer Communication and Networks (ICCCN), 2014.

[14]. S. Wang, L. Minku and X. Yao, "A Learning Framework for online class imbalance Learning," Proc. IEEE symposium Computational Intelligence and Ensemble Learning. pp 36-45, 2013.

[15]. S. Wang, L. Minku and X. Yao, "Resampling-Based Ensemble Methods for Online Class Imbalanced Learning," IEEE Transactions on Knowledge and Data Engineering. Vol 27, No 5, May 2015. PP 1356-1368

[16]. C. Seiffert, T. Khoshgoftaar, J. Hulse and A. Napolitano. "RUSBoost: A Hybrid Approach to Alleviating Class imbalance," IEEE Transactions on Systems, Man, and Cybernatics-Part A: Systems and Humans, Vol 40, No 1, Jan 2010.

[17]. Oza N., Russell S., "Experimental comparisons of online and batch versions of bagging and boosting", In Proceedings of the seventh ACM SIGKDD international conference on Knowledge discovery and data mining (KDD'01), pp 359-364, 2001.