Review on Class Imbalance Learning: Binary and Multiclass

Ranjana Singh
ME, Student
Dr. D Y Patil School of Engineering and Technology
Savitribai Phule Pune University, Pune

Roshani Raut (Ade)
Assistant Professor
Dr. D Y Patil School of Engineering and Technology
Savitribai Phule Pune University, Pune

ABSTRACT

The application area of technology is expanding the span of information size is also additionally increases. Classification gets to be troublesome in view of unbounded size and imbalance nature of data. Class imbalance where one of the two classes having more sample than other years. There are typical strategies for an imbalance data set which is zoned into three main categories, the algorithmic methodology, data pre-processing approach and feature selection approach. In this paper every methodology is characterize which gives the right bearing for exploration in the class imbalance problem. This Paper also examines the three basic divisions of class Imbalance learning like data-preprocessing, the algorithmic approach, and feature selection approach.

General Terms

Class Imbalance Learning

Keywords

Machine learning, Imbalanced Data, Binary Classification, Multiclass Classification, Dynamic Sampling

1. INTRODUCTION

Defect prediction in software is one of the most cost effective operations. Software practitioners see it was an essential stage on which the character of the software being produced depends. It has got a major piece in bringing down the allegations on the product business, of being unable to convey the requirements at the developer of the spending plan and on time. Besides this, the clients' response regarding the product quality has shown a great shift from unsatisfactory to satisfactory.

A wide range of machine learning techniques has been studied to facilitate software testing, and save testing costs in software modules. The imbalanced nature of this sort of data makes the learning problem of such projects. Problem with imbalanced distributions is successfully tackled by Class imbalance learnings [1]. Numerous traditional algorithms for machine learnings furthermore, data mining issues except that the objective classes offer comparative earlier probabilities.

Support Vector Machines are popular machine learning method. Requiring consideration of all its theoretical and practical advantages, SVMs could create ideal results with imbalanced datasets [2]. In machine learning, the group that is ensemble of classifiers is known not the accuracy single classifiers by collecting at several of them [3].

Binary-class events is the main focus of the existing research focused only on. Imbalanced class distribution may take place in many arenas, such as network intrusion detection [4], financial engineering [5], and medical diagnostics [6]. The classifiers tend to produce high classification accuracies on the majority classes, but poor classification accuracies on the minority ones [7]. Class imbalance learning specializes in tackling classification problems with imbalanced distributions, which could be helpful for defect prediction.

Software Defect Prediction (SDP) is unitary of the significant activities during the Testing Phase of SDLC. It places the mental faculties that are hard and require extended testing. The testing assets can be used productively without breaking the constraints. Despite the fact that SDP is extremely useful in testing, it's not generally simple to predict the faulty modules. On that point are several issues that impede the smooth carrying out every bit well as utilization of the Defect Prediction models. In this story, the authors have identified some of the significant issues of SDP and concentrated on what has been reached out thus far to address them [8].

Class imbalance learning refers to a sort of classification issues, where some years are highly underrepresented compared to other years. The skewed distribution makes many conventional machine learning algorithms less effective, particularly in predicting minority class examples.

Numerous amount of study has been performed in the area of software defect prediction involving a varied routine of techniques such as bagging, boosting, naive Bayes, one ruler, and so on [9]. A brief recap of the significant work carried out in this study has been provided by Arora and Saha [10].

2. SOFTWARE DEFECT PREDICTION: ISSUES

As today's software grows in size and complexity, how to sustain the high caliber of the product is one of the most significant problems facing the software manufacture. Software Defect Prediction (SDP) can be made up as a learning problem in software engineering, which has attracted growing interest from both academe and industry. Static code attributes are pulled from previous releases of the software with the log files of defects, and drilled to build models to predict defective modules for the accompanying press release. It serves to turn up. Portions of the software those are more probable to control faults.

This endeavor is particularly useful when the project budget is put up, or the whole software system is too large to be tested thoroughly. A good defect predictor can guide software engineers to focus the testing on defect-prone sections of the software. For a high-performance defect predictor, researchers have been exercising on the choice of suitable properties and effective learning algorithms. Several statistical and machine learning methods have been investigated for SDP, among which Naive Bayes and Random Forest. This part presents the troubles faced in software defect prediction and the solutions offered by the eminent researchers for these troubles. It also addresses the unresolved issues in this region as well [11], they are as follows

1. Relationship between Attributes and Fault
2. No Standard Measures for Performance Assessment
3. Issues with Cross-Project Defect Prediction
4. No General Framework Available

3. CLASS IMBALANCE PROBLEM

The class distribution of the training data determines the efficiency of Software Fault prediction models [12]. The Class distribution is defined as, as the number of instances of each form of educational activity in the training data set. If the number of instances belonging to one class is much more than the number of examples belonging to another stratum, and then the problem is known as class imbalance problem [13]. The grade with more instances is called majority class and the one with lesser instances is called minority class. The problem widens when the class under consideration, i.e. the wrong year is represented by fewer instances. Several techniques have been suggested for addressing this problem.

Barandela et al [14] carried out a comparative field of various sampling, i.e. resizing techniques including Undersampling and Oversampling techniques for handling class imbalance and analyzed the comparative performance of these two categories of sampling techniques. They concluded that, in case of highly imbalanced data sets, oversampling of minority class should be caused, whereas, if the datasets are not severely biased then undersampling is better. Also, the combination of Undersampling and Oversampling can be a more authentic option. An empirical study was held out by Zhou and Liu for studying the outcome of sampling and threshold moving on the training of cost-sensitive neural networks. The answers demonstrated that cost sensitive learning is easy for binary class dataset and difficult for multi-class dataset as well as for a highly imbalanced dataset.

The sampling technique used was Synthetic Minority Oversampling Technique (SMOTE) and the classifier used was Decision Tree Classifier. Khoshgoftaar et al. [15] compared 5 data sampling techniques, namely, Random Under Sampling (RUS), SMOTE, Borderline-SMOTE, Random, Over Sampling (ROS) and Wilson's Editing (WE) with Boosting algorithm and concluded that sampling performs significantly well but Boosting performs even more unspoilt.

A Genetic Algorithm based sampling technique called Evolutionary Sampling was proposed by Khoshgoftaar et al and compared with various imbalances handling methods using two classifiers namely C4.5 and RIPPER. The proposed technique proved to be more serious than the other techniques. Khoshgoftaar et al. [16] offered a hybrid Sampling/Boosting algorithm to address class imbalance namely RUSBoost.

The taking out of the proposed algorithm was then compared to its individual techniques, namely RUS and Boosting as well as with another hybrid Sampling Boosting algorithm SMOTEBoost and its individual techniques i.e. SMOTE. It was concluded that, the operation of two hybrid algorithms is not significantly different, but RUSBoost is simpler and quicker than the other.

Fuller et al. [17] proposed a taxonomy of ensemble based algorithms for addressing class imbalance. They also showed empirically that ensemble methods, improve the performance of prediction models as compared to preprocessing techniques applied to a single classifier model. Wang and Yao [18] investigated if class imbalance learning can benefit software defect prediction and if yes, then how. They compared various imbalance learning methods, namely RUS, Balanced RUS, threshold moving, SMOTEBoost and AdaBoost.NC, and took note that the best performance was achieved using AdaBoost.NC.

4. APPROACHES

Several techniques have been aimed to resolve the troubles associated with class imbalance, which divided into three basic categories data-preprocessing, the algorithmic approach, and feature selection approach.

4.1 Data-Preprocessing

Please In data-preprocessing technique, sampling is applied to data in which either new samples are added or existing samples are transferred. The procedure of adding new sample in existing is known as over-sampling and process of removing a sample known as under-sampling. Data level methods for balancing the classes consists of resembling the original data set, either by over-sampling the minority class or by under-sampling and/or under-sampling the majority class, until the classes are about equally represented [19].

4.2 The Algorithmic Approach

A several new algorithms have been made for solving the class imbalance problem. The finish of this approach is to optimize the performance of learning algorithm on unseen data. Single-class learning methods recognized the sample belongs to that year and reject others. Under certain condition such as multi-dimensional data set one class learning gives better performance than others [20].

5. BINARY CLASS IMBALANCE

Class imbalance learning is a growing research area in machine learning that trains to better deal with this sort of trouble. Generally speaking, a pre-sampling method makes the training set balanced, either by oversampling the minority class or buy under sampling the majority class. Besides pre-sampling methods, cost-sensitive methods are also regarded as important approaches to class imbalance problems. The main idea is that, to ward away the minority class being overlooked, a higher misclassification cost should be attributed to it than to the majority class. In this way, a class imbalance problem can be formulated as a cost sensitive learning problem and estimated out by an existing method.

Boosting and active learning methods, though not immediately motivated by class imbalance problems, actually integrate the sampling and training operation. Hence, they both have been proposed to tackle class imbalance problems. Boosting type methods, such as SMOTEBoost (SMB), random undersampling (RUS-ball). One major challenge of using class imbalance learning methods is how to choose appropriate parameters, such as the sampling rate, and misclassification cost of classes, which are essential to their inductive reasoning of the minority class, and can be time-consuming and problem-dependent to tune.

Most of the above methods were primarily developed for binary-class problems, while their efficiency on multiclass problems have not been well investigated. On one hand, it is nontrivial to extend boosting-type and active learning methods to multiclass problems. On the other hand, trying out and cost sensitive methods are readily applicable for both binary class and multiclass problem. Nevertheless, it is not easy to pre-determine the sampling ratio or misclassification cost for each stratum. In fact, empirical study has indicated that most sampling methods are ineffective on multiclass problems and often induce a negative result of the troubles with big number of categories.

Binary classifiers have typically been the norm for building classification models in the Machine Learning community. Even so, an alternate to binary classification is one-class categorization, which aims to build models using only a single class of information. This is particularly useful when there is an overabundance of data on a special path of study.

6. MULTICLASS IMBALANCE LEARNING

Boosting Two types of multiclass could occur in an imbalanced data set: one majority and multiple minority classes (multi minority cases), and one minority and multiple majority classes (multi majority cases). A problem with multiple minority and multiple majority classes can be treated as the instance when both types occur.

Several interesting research questions are set up here: Is there any difference between multiple minority and multiple majority classes? Is the problem posing the same or different challenges to a learning algorithm? Which one would be more difficult to tackle? Which aspects of a problem would be affected the most by the meticulous? Would it be a minority class, a majority class or both?

Most of the above methods were primarily developed for binary-class problems, while their efficacies on multiclass problems have not been well investigated. On one hand, it is nontrivial to extend boosting-type and active learning methods to multiclass problems. On the other hand, testing out and cost sensitive methods are readily applicable for both binary class and multiclass problem. However, it is not easy to pre-set the sampling ratio or misclassification cost for each stratum. In fact, empirical study has shown that most sampling methods are ineffective on multiclass problems and frequently cause a negative consequence of the problems with large number of classes.

Pre-sampling methods focus on dealing with the training set before training. The aim is to bring to the training set balanced or less imbalanced by adding examples to or removing instances from the training set before training the classifier. ROS and RUS are two basic sampling methods [21] which examples are randomly duplicated ROS or removed RUS to make the training set balanced. ROS and RUS can also be employed to a training set simultaneously. For multiclass problems, ROS and RUS can be simply implemented [22].

On that point are some papers about resampling techniques that study the effect of the changing class distribution to deal with imbalanced datasets, where it has been empirically shown that the application of a preprocessing step in parliamentary procedure to balance the class distribution is usually a positive solution [23].

Among the available class imbalance learning techniques, random under sampling and oversampling are simple yet very popular resampling methods. At random under sampling, the sheaths of the majority class are removed until the datasets are balanced. The minority class examples are randomly duplicated to balance the datasets called at random oversampling [24].

Although the written reports on cost-sensitive problems considered both binary-class and meticulous cases, there are only a few methods presented for developing a proper cost matrix for a class imbalance problem. Japkowicz and Stephen [25] modified the misclassification cost to build up for the imbalance ratio of the classes in their experiments on binary-class problems.

A SMOTE generated the same scrap of synthetic models for each minority example and this strategy may cause data overlapping [26]. Some methods, which can overcome this limitation of SMOTE have been proposed such as borderline SMOTE [27] and Adasyn [28].

7. DYNAMIC SAMPLING

Dynamic Sampling (DyS) dynamically selects examples for training during the training process. No one at once pre-delete any instance to prevent information loss, and one should dynamically select examples for training to avoid redundant information and to make the best use of the training data. Specifically, in the sequential mode for training MLP, the training instances are used to update the MLP one by one. Unlike the pre-sampling and cost-sensitive methods, DyS integrates the sampling and training procedures as a whole. Different from boosting-type methods, which sequentially train a number of base classifiers, DyS continuously updates the weights of a single MLP. In comparing two active learning methods, dish makes use of the label and all the features of an object lesson to mark whether the example should be insured, while the label information is not utilized in a standard active learning context [1].

8. COMPARITIVE STUDY

In some previous research work [24], it was found that the “random oversampling + AdaBoost.NC” tree ensemble is effective in handling two-class imbalance problems. It presents a good recognition rate of the minority class and balances the performance between minority and majority classes well by making use of ensemble diversity. Moreover, this training strategy is flexible and simple without taking out any training data. For the aforesaid reasons, researchers look into this algorithm and continue on with their work of meticulous cases in this part. The primary research question here is whether AdaBoost.NC is still efficient in solving multiclass imbalance problems. In order to answer the question and to find out if class decomposition is necessary, AdaBoost.NC is compared with other state-of-the-art methods in cases of using and not using class decomposition, including the conventional AdaBoost, resampling-based AdaBoost, and SMOTEBoost. AdaBoost is discussed as the baseline method, because the AdaBoost.NC algorithm is in the boosting training framework. Resampling techniques and the SMOTEBoost algorithm are examined for their extensive usage in the meticulous imbalance learning literature.

Dynamic Sampling (DyS) manages to dynamically choose the training data to be utilized in each epoch of MLP. Further, this proposed algorithm is compared with Adaboost.NC algorithm that traverses over the parameter setting issue. From comparative analysis, it is found that the proposed algorithm gives good effects as compare to Adabbost.NC. Analysis drawn from a comparative field of each of the algorithm is presented in the following table.

Algorithm	Advantages	Disadvantages
AdaBoost.NC	This is a standard technique for improving prediction accuracy Minority	Ignore overall performance of Classifier
RUSBoost	Simpler, faster and less complex than SMOTE Boost algorithm	Unable to solve Multiclass Imbalance problem
Dynamic Sampling MLP	DyS integrates the sampling and training processes as a whole.	Ignore imbalance class distribution.

9. CONCLUSION

Data preprocessing provide better resolution than other methods because it allows adding new information or deleting the superfluous data, which helps to balance the data. Some other method that helpful to solve the problem of class imbalance is boosting. Supercharging is a powerful ensemble learning algorithm that improved the performance of weak classifier. The feature selection method can also be applied for classification of imbalance data. A training algorithm of MLPs for multiclass imbalance issues with a simple yet effective dynamic sampling method, DyS, has been analysed can be applied for multiclass imbalance. In every epoch of the training procedure, a probability was predictable for each sample provide to the MLP.

DyS can outperform additional appropriate methods, including pre-sample approaches, active learning approaches, cost-sensitive approaches, and boosting type approaches...for multiclass imbalance issues

10. REFERENCES

[1] Shuo Wang, Member, IEEE, and Xin Yao, “Using Class Imbalance Learning for Software Defect Prediction”, IEEE TRANSACTIONS ON RELIABILITY, VOL. 62, NO. 2, JUNE 2013.

[2] Haibo He and Yunqian Ma, “Imbalanced Learning: Foundations, Algorithms, and Applications.”

[3] Mikel Galar, Alberto Fern´andez, Edurne Barrenechea, Humberto Bustince, “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

[4] J. Zhang and M. Zulkernine, “Network intrusion detection using random forests,” in Proc. 3rd Annu. Conf. Privacy, Secur. Trust, 2005, pp. 53–61.

[5] G. Wang, “Asymmetric random subspace method for imbalanced credit risk evaluation,” in Proc. Softw. Eng. Knowl. Eng., Theory Pract., 2012, pp. 1047–1053.

[6] M. A. Mazurowski, P. A. Habas, J. M. Zurada, J. Y. Lo, J. A. Baker, and G. D. Tourassi, “Training neural network classifiers for medical decision making: The effects of imbalanced datasets on classificationperformance,” Neural Netw., vol. 21, nos. 2–3, pp. 427–436, 2008.

[7] Minlong Lin, Ke Tang, . ”Dynamic Sampling Approach to Training Neural Networks for Multiclass Imbalance Classification”, IEEE TRANSACTIONS ON NEURAL NETWORKS AND LEARNING SYSTEMS, VOL. 24, NO. 4, APRIL 2013.

[8] Ishani Aroraa , Vivek Tetarwala,*, Anju Sahaa, “Open Issues in Software Defect Prediction”, International Conference on Information and Communication Technologies (ICICT 2014), Elsevier, Procedia Computer Science 46 (2015) 906 – 912.

[9] Jiang Y, Lin J, Cukic B, Menzies, T. Variance analysis in software fault prediction models. In: 20th International Symposium on Software Reliability Engineering. Mysuru; 2009. p. 99-108.

[10] Arora I, Saha A. A literature review on software defect prediction. In: Second International Conference on Emerging Research in Computing, Information, Communication and Applications. Bangalore; 2014. p. 478-487.

[11] Batista GEAPA, Prati RC, Monard MC. A study of the behavior of several methods for balancing machine learning training data. SIGKDD Explorations 2004; 6:20-9.

[12] Galar M, Fernandez A, Barrenechea E, Bustince H, Herrera F. 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 2012; 42:463-84.

[13] Barandela R, Valdovinos RM, Sánchez JS, Ferri FJ. The imbalanced training sample problem: Under or over sampling? Structural, Syntactic, and Statistical Pattern Recognition. Springer Berlin Heidelberg; 2004. p. 806-814.

[14] Zhou ZH, Liu XY. Training cost-sensitive neural networks with methods addressing the class imbalance problem. IEEE Transactions on Knowledge and Data Engineering 2006; 18:63-77.

[15] Seiffert C, Khoshgoftaar TM, Van Hulse J, Napolitano A. Building useful models from imbalanced data with sampling and boosting. In: FLAIRS Conference. 2008. p. 306-311.

[16] .Seiffert C, Khoshgoftaar TM, Van Hulse J, Napolitano A. RUSBoost: A hybrid approach to alleviating class imbalance. IEEE Transactions on Systems, Man and Cybernetics, Part A: Systems and Humans 2010; 40:185-97.

[17] Galar M, Fernandez A, Barrenechea E, Bustince H, Herrera F. 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 2012; 42:463-84.

[18] Wang S, Minku LL, Yao X. A learning framework for online class imbalance learning. In: IEEE Symposium on Computational Intelligence and Ensemble Learning (CIEL). 2013. p. 36-45.

[19] V. García J.S. Sánchez R.A. Mollineda R. Alejo J.M. Sotoca, "The class imbalance problem in pattern classification and learning", Pattern Analysis and Learning Group.

[20] Mike Wasikowski, Member and Xue-wen Chen, "Combating the Small Sample Class Imbalance Problem Using FeatureSelection", IEEE Transactions on Knowledge and Data Engineering, Vol. 22, No. 10, October 2010.

[21] Peters F, Menzies T, Gong L, Zhang H. Balancing privacy and utility in cross-company defect prediction. IEEE Transactions on Software Engineering 2013; 39:1054-68.

[22] S. Wang and X. Yao, "Multiclass imbalance problems: Analysis and potential solutions," IEEE Trans. Syst., Man Cybern. B, vol. 42, no. 4, pp. 1119–1130, Apr. 2012.

[23] G. E. A. P. A. Batista, R. C. Prati, and M. C. Monard, "A study of the behavior of several methods for balancing machine learning training data," SIGKDD Expl. Newslett., vol. 6, pp. 20–29, 2004.

[24] Y. Sun, M. S. Kamel, and Y. Wang, "Boosting for learning multiple classes with imbalanced class distribution," in Proc. Int. Conf. Data Min., 2006, pp. 592–602.

[25] N. V. Chawla, A. Lazarevic, L. O. Hall, and K. W. Bowyer, "SMOTEBoost: Improving prediction of the minority class in boosting," in Proc. Knowl. Dis. Databas. Conf., 2003, pp. 107–119.

[26] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer, "SMOTE: Synthetic minority over-sampling technique," J. Artif. Intell. Res., vol. 16, no. 1, pp. 321–357, 2002.

[27] H. Han, W. Y. Wang, and B. H. Mao, "Borderline-SMOTE: A new oversampling method in imbalanced data sets learning," Adv. Intell. Comput.,vol. 2, no. 5, pp. 878–887, 2005.

[28] 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 www.ijcsn.org ISSN 2277-5420