GE-International Journal of Engineering Research
Vol. 4, Issue 8, August 2016 IF- 4.721 ISSN: (2321-1717)

Website: www.aarf.asia Email : editor@aarf.asia , editoraarf@gmail.com

PREDICTING SEISMIC BUMPS BY USING CLASSIFIER ENSEMBLES

Sami Halawani

Faculty of Computing and Information Technology, King Abdulaziz University, Jeddah, Saudi Arabia.

ABSTRACT

Predicting seismic bums is an important research area in mining. As correct prediction save many human lives, attempts are being made for the correct prediction of seismic bums. In this paper, we use many machine learning techniques to predict seismic bums. Results suggest that classifier ensembles generally do not perform well than single classifiers. As the dataset is imbalanced machine learning techniques developed for imbalanced datasets may be more useful for seismic mass predictions.

KEYWORDS - Seismic mass, Classifiers, Ensembles, Imbalanced dataset.

1. INTRODUCTION

Mining activity is an important part of our life. However, mining activities have their hazards effects. Early prediction of the seismic activities may reduce the dangerous effects of seismic activities and can save the human lives [1]. However, it is not simple to predict these activities. The task is to predict whether a notable seismic activity will take place or not.

Machine learning techniques are powerful tools to analyze the data [2]. These techniques are also used to predict the future events. Classification is an important machine learning technique. In this technique first a model is trained on a given dataset and then a model is used to predict whether an outcome will take place or not. This problem is called binary classification. Decision trees, neural networks, support vector machines etc. are popular classification tools [2].

Classifier ensembles are mixture of many classifiers. Each classifier contributes to the final result of an ensemble. These ensembles have shown better classification accuracy than single classifiers [3].

In this paper we will use the ensembles of various classifiers to predict the notable seismic activity. The paper is organized in following way. Section 2 has discussion about the data set and machine learning techniques used in this paper. Section 3 has results and discussion. Section 4 has conclusion and future work.

2. MATERIAL AND METHODS

This dataset is collected from Polish mines [1]. The dataset has 2584 data points and 18 attributes. Most of these attribute have the information about the previous shift. The output has two classes. The dataset has only 170 positive data points. The task is to predict whether the next day there will be seismic bumps or not. High energy (higher than 10^4 J) is defined as the targeted seismic bumps. The information about the attributes is given in Table 1.

Table 1- Information about dataset attributes

Attributes	Information
1	Seismic: result of shift seismic hazard assessment
2	Seismoacoustic: result of shift seismic hazard assessment
3	seismic energy of previous shift
4	a number of pulses of previous shift
5	a deviation of energy of previous shift
6	a deviation of energy of previous shift
7	a deviation of a number of pulses of previous shift
8	Result of shift seismic hazard assessment in the mine working
9-16	The number of bums of various energy in previous shift
17	Total seismic energy of the previous shift.
18	Maxenergy: the maximum energy of the seismic bumps

2.1 Machine Learning Tools

Following machine learning techniques (classification methods) are used for the prediction.

2.1.1 Decision Trees - Decision trees are very popular classifier [4, 5]. It is a rule based method. It is very popular because of its low computational complexity and good interpretability. The rules are easily understandable to human beings. While growing a decision tree at each node available data points are divided into two subsets depending upon the importance of attributes.

2.1.2 Naïve Bayes – Naïve Bayes classifier is a probabilistic classifier [2]. The assumption is that all attributes are independent. However, these classifiers are very accurate.

2.1.3- Neural networks - These neural networks mimic the human brains [2]. These consist of interconnected nodes. These neural networks can approximate nonlinear decision boundaries.

2.1.4- Support vector machines – Support vector machines are very popular classifiers [2]. With proper kernel functions they can easily represent nonlinear decision boundaries. However, if proper kernel functions are not selected it can give inaccurate results.

2.1.5 Classifier Ensembles – Classifier ensembles are combination of many classifiers [3, 6-9]. Each classifier contributes to the final result. The result of an ensemble is generally more accurate than a member classifier. The condition for an accurate ensemble is that member classifiers of the ensemble are accurate and diverse. There are many methods to create accurate and diverse classifiers. Bagging and Boosting are the two most popular ensemble methods. We will use these two methods in this paper. These two methods will be discussed in detail.

2.1.5.1 Bagging – In bagging diverse datasets are created [6]. These datasets are created by random sampling with replacement. Many datasets are created by using the same method. As these datasets have different data points these datasets are diverse. Classifiers trained on these diverse datasets are diverse so they create an ensemble.

2.1.5.2- Boosting - In boosting, classifier are trained one by one [7]. Each data points have equal weights. One classifier is trained on this dataset. The weights of the points which are wrongly classified by the classifier are increased so in the next round these points have more probability to be selected. All classifiers are trained by using this method. In subsequent round classifiers concentrate more on hard to predict points. The boosting gives accurate results however when datasets have noisy points it can produce poor results as in subsequent rounds classifiers will concentrate more on noisy data points.

3. RESULTS AND DISCUSSION

We used WEKA software for our experiments [10]. J48 decision trees were used in our experiments [4]. All the defaults values were used in different methods. The size of the ensemble was 10 for all ensembles. Ten folds cross validation was used. The results are presented in Table 2 – Table 5.

Table 2- Results With Decision Trees

Name of the method	Accuracy	Precision	Recall
Single decision tree	93.34 %	0.873	0.933
Decision tree ensembles with Bagging	92.80 %	0.892	0.928
Decision trees ensembles with Boosting	91.33 %	0.891	0.913

Table 3- Results With Naïve Bayes Classifier

Name of the method	Accuracy	Precision	Recall
Single Naïve Bayes	86.27 %	0.907	0.867
Naïve Bayes ensembles with Bagging	89.99 %	0.907	0.866
Naive Bayes ensembles with Boosting	87.13 %	0.905	0.871

Table 4- Results with Neural Networks

Name of the method	Accuracy	Precision	Recall
Single Neural Network	92.45 %	0.891	0.925
Neural Network ensembles with Bagging	93.29 %	0.913	0.928
Neural Network ensembles with Boosting	93.24 %	0.925	0.932

Table 5- Results with Support Vector Machines

Name of the method	Accuracy	Precision	Recall
Single Support Vector Machine	93.42 %	0.873	0.934
Support Vector Machine ensembles with Bagging	93.42 %	0.873	0.934
Support Vector Machine ensembles with Boosting	93.42 %	0.873	0.934

The results suggest that the ensemble methods do not improve a lot against single classifier. For example the accuracy of single decision tree is better than ensemble of decision trees whereas ensemble of decision trees have better precision than single decision tree. For naïve Bayes classifier the ensembles of Naive Bayes classifiers have better accuracy than single Naïve Bayes classifiers. However, ensembles of Naïve Bayes classifiers do not show any improvement in precision. For neural networks ensembles there is a small improvement in accuracy and precision over single neural networks. For support vector machines classifiers, ensembles and single classifier perform exactly the same. The reason for this is that all the data appoints are classified in one class. This shows that support vector machine is not a good classifier for this problem.

The dataset is an unbalanced dataset in which points of one class is very small. This could be a reason for poor performance of support vector machines.

4. CONCLUSION

In this paper, we studied predicting seismic bumps by using various machine learning techniques. Single classifier and ensembles of classifiers were used for this purpose. Accuracy, precision and recall performance measures were used in our experiments. Results suggest that ensembles are not very useful for this application. Support vector machines also did not give good results as it put all the data points in one class.

As this data in unbalanced dataset, in future we will use the methods that have been developed for these kinds of datasets.

REFERENCES

1. Sikora M., Wrobel L.: Application of rule induction algorithms for analysis of data collected by seismic hazard monitoring systems in coal mines. Archives of Mining Sciences, 55(1), 2010, 91-114.
2. Han, J; Kamber, M;, Pei, J; Data Mining: Concepts and Techniques, Morgan Kaufmann, 2011.
3. Kuncheva, L. I., Combining pattern classifiers: methods and algorithms. Wiley-IEEE Press, New York, 2004.
4. Quinlan, J. R., C4.5: Programs for machine learning. CA: Morgan Kaufmann, San Mateo, 1993.
5. Breiman, L., J. Friedman, R. Olshen and C. Stone, Classification and Regression Trees. 1st Edn., Chapman and Hall, London, ISBN- 10: 0412048418, 1984.
6. Breiman, L., Bagging predictors. Mach. Learn. 24(2):123–140, 1996.
7. Freund, Y. and R.E. Schapire; A decision theoretic generalization of on-line learning and an application to boosting. J. Comput. Syst. Sci., 1997, 55: 119-139. DOI: 10.1006/jcss.1997.1504
8. Ho, T. K.; The Random Subspace Method for Constructing Decision Forests, IEEE Transactions on Pattern Analysis and Machine Intelligence, 1998, 20, no. 8, 832–844. DOI:10.1109/34.709601
9. Breiman, L., Random Forests, Machine Learning, 2001, 45, no. 1, 5–32. DOI:10.1023/A:1010933404324.
10. Witten, I. H., and Frank, E., Data mining: practical machine learning tools with java implementations. Morgan Kaufmann, San Francisco, 2000.