IOP Conf. Series: Materials Science and Engineering **428** (2018) 012004 doi:10.1088/1757-899X/428/1/012004

Bagging of Xgboost Classifiers with Random Under-sampling and Tomek Link for Noisy Label-imbalanced Data

Luo Ruisen[1], Dian Songyi[1], Wang Chen[1, 2 *], Cheng Peng[1], Tang Zuodong[1], Yu YanMei[1] and Wang Shixiong[3]

[1] College of Electrical Engineering and Information Technology, Sichuan University, 24 South Section 1, One Ring Road, Chengdu, China, 610065
[2] Department of Computer Science, University College London, Gower Street, London, United Kingdom, WC1E 6BT
[3] School of Electronics and Information, Northwestern Polytechnical University, 1 Dongxiang Road, Chang'an District, Xi'an, China, 710129

[*]chen.wang@ucl.ac.uk

Abstract. Fitting label-imbalanced data with high level of noise is one of the major challenges in learning-based intelligent system design. In this paper, for the two-class problem, we propose a bagging-based algorithm with Xgboost classifier (Gradient Boosting Machine) and under-sampling approaches to overcome the challenge. To avoid model misspecification caused by imbalanced data, random sampling with replacement is employed to obtain several balanced training sets; and to mitigate the problem of misleading information produced by noise, Tomek Link method is introduced to eliminate the cross-class overlapped instances, which are the primal sources of noise. And to obtain robust individual learners, we utilize Xgboost, a novel Gradient Boosting Machine-based classifier with convenient parameter tuning interface, to fit each component of the bagging ensemble. The performance of the proposed method is tested with Mandarin radio records (MFCC features) with the task of keywords recognition, and experimental results show that the new method could outperform single Xgboost classifier, verified the rationality and effectiveness of the bagging scheme. The method proposed in the paper could offer a novel solution to the challenge of noisy imbalanced data classification, and the implementation of Xgboost in this area could also serve as an innovative work.

1. Introduction

Data-driven learning algorithms are grasping increasing popularities in the designing of automation and intelligent systems with their record-breaking capacities for various tasks [1]. While this branch of algorithms has shown state-of-the-art performances, learning with highly-noisy class-imbalanced data imposes a significant challenge for the further development of them. Specifically, for classification tasks in intelligent and automation system designing, most of the learning algorithms are based on the assumption of a roughly class-balanced dataset with minor noise; in practice, however, data collected from industrial applications could be highly skewed in label distribution and associated with significant noise. Thus, when encountering label-skewed noisy data in automation and intelligent system, the performances of the advanced learning techniques often drop drastically [2].

There exist previous research endeavours to address the problem, and they could be roughly categorized into four types: surrogate loss functions, re-sampling methods, ensemble learning and special class feature representation (so-called one-class learning) [3]. Among the above categories, the re-sampling approaches, which intend to produce a balanced dataset by oversampling the minority class and/or sub-sampling the majority class, and ensemble learning methods, which aims to stack

multiple learners working together to get a more satisfying prediction, could be regarded as approaches from a data-level perspective. The major advantage of these two types of methods is that it does not demand the modification of the structure of the classifiers, thus they could be flexibly plugged into different types of powerful individual algorithms. Previous research mostly utilizes re-sampling and ensemble method separately, and it naturally comes to an intuition that a hybrid of the two methods might have a better performance, which constitute the inspiration of this paper.

In this paper, following the above inspirations of sub-sampling and ensemble learning, we design a novel algorithm to process class-imbalanced noisy data for two-class classification. Specifically, for a dataset considered as label unbalanced and significantly noisy-affected, the algorithm will firstly produce several balanced sub-datasets by randomly sampling the majority with replacement and combining them with the minority. The number of sub-datasets are specified manually, and this process this much alike conventional bagging method. After obtaining the datasets, we perform sub-sampling method, more specifically Tomek Link elimination [4], on them to mitigate the noise by removing the overlapped data between the two classes. Subsequently, we could fit one classifier per subset to get a bagging algorithm result. In our method, the individual classifier is selected as Xgboost Classifier, which is an advanced implementation of the popular-employed Gradient Boosting Machine [5]. And finally, when predicting a new example, the algorithm could weight each individual voter equally and output a probability, which will be its confidence in predicting the instance as majority/minority.

To this end, the rest of the paper is arranged as follows: section 2 will be reviewing related work in the area and discussing their relationships with our work; section 3 will be adopted as detailed introduction of our proposed algorithm; the experimental results will be illustrated in section 4 together with the setting of the experiment and the discussions regarding the result; and finally, a general conclusion will be demonstrated in section 5.

2. Related Work

Re-sampling has been served as one of the conventional approaches to process imbalance data classification [6], and major publications have discussed both over-sampling and under-sampling algorithms. However, both over- and under- sampling methods could incorporate bias to the data and thus adversely affect the results [7]. And despite the abundant number of publications, it still remains unclear whether over- or under- sampling will be more preferable regarding the overall performance [8]. In this paper, under-sampling is considered for the purpose of ensemble. Popular under-sampling algorithms include NearMiss [9], Condensed Nearest Neighbour [10], as well as the Tomek Link method employed in our algorithm. [11] summarized various under-sampling methods and provided related APIs, which are adopted in the program of the proposed algorithm. Notice that although all the algorithms in this branch are all labelled as ‘under-sampling’, the intuitions behind them could vary to a large extent. For instance, [12] method aims to produce several distinct representations for the majority class, while Tomek Link, the method utilized in our algorithm, primarily serves as an algorithm to reduce noise.

In addition to re-sampling, ensemble learning has also been adopted as an effective method in tackling label-skewed data classification [13]. Both bagging and boosting ensemble algorithms are included in previous literatures [14], and it is noteworthy that when training a bagging ensemble model on label-skewed data, the requirement of random sampling with replacement could be regarded as a form of under-sampling. And if we only under-sample the majority class, it will be possible to regard the under-sampling techniques as part of ensemble and merge the two branches of approaches. [15] elaborated on the above idea and argued that simply combining random under-sampling and bagging could achieve a satisfying performance. However, the authors did not consider the situation when the noise level is high. [16] discussed a combined method of an over-sampling strategy with bagging of SVMs, which is similar to our approach but focusing on another aspect of re-sampling.

Regarding the Xgboost classifier utilized in our algorithm, there also exists plentiful literatures discussing the technique. The classifier is based on Gradient Boosting Machine and it was firstly

introduced by the Chen et al. in 2016 [5]. It illustrates impressive performances on various Machine Learning challenges [17] and has been implemented to different scientific and engineering sectors, including predictive finance [18], symptom analysis [19] and biostatistics [20]. However, to the best of the authors' knowledge, there has not been any research outcomes focusing on the bagging of Xgboost models to get better predictions. As a boosting ensemble method, Xgboost tends to more likely to be overfitting [21], thus it is intuitive to think that applying bagging beyond Xgboost models could improve the overall performance. Meanwhile, by removing the overlapped instances with Tomek Link, we could maximize the advantage of reducing bias in boosting while avoid potential overfitting of noise.

3. Methodology

The holistic algorithm is composed of four major parts: random sampling with replacement of the majority data, Tomek Link elimination, individual Xgboost classifier fitting, and bagging of different classifiers for prediction. In this section, the details of each part of the algorithm will be discussed and we will show the intuitions and rationalities of the adopted techniques.

3.1. Random Sampling with Replacement

Random sampling with replacement is the standard technique utilized in bagging algorithm to produce sub-datasets, and the difference in our algorithm is that we only perform this operation on the majority data, which constitute our primal under-sampling algorithm. In our method, to avoid sampling the same instance for multiple times in one subset, we directly sample p distinct instance by getting a random permutation of the index of the majority dataset, and select the first p instances of the permutated data.

Random sampling with replacement could produce datasets sufficiently representing the variance of the data. On the contrary, more complicated under-sampling methods, such as clustering-based technique, tend to better represent the bias which we demand the classifiers to learn. In our algorithm, since we would like to focus on reducing variance during the bagging procedure, it is preferable to directly utilize random sampling with replacement. The comparison between simple random sampling with replacement and the complex clustering-based under-sampling could be shown in figure 1.

Since we intend to obtain a set of instance-balanced data, ideally, we would set the sampling size $p = m_k$, where m_k is the size of the data lying in minority class. However, as we will see in the following paragraphs, the randomly sampled data will be further processed with Tomek Link Elimination, which means, the instances of majority which overlapped with the minorities will be removed. Thus, we would like to add some additional instances to the majority class. In our algorithm, we set $p = \lfloor 1.5 * m_k \rfloor$.

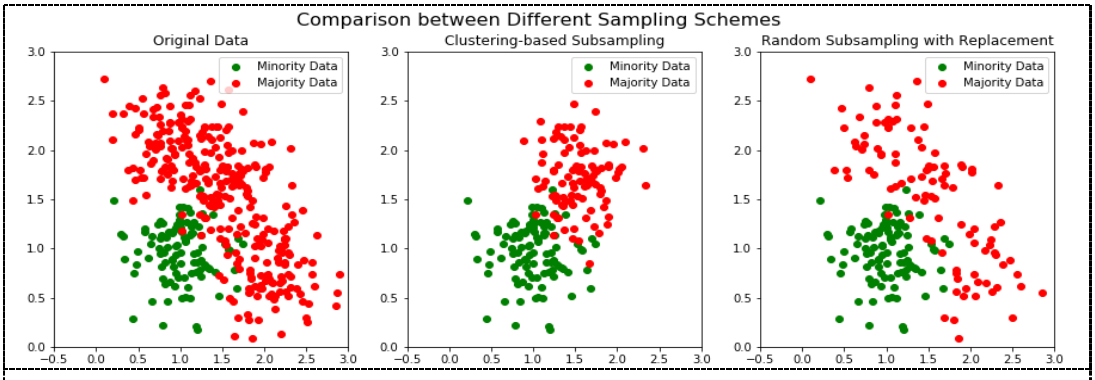

Figure 1. Comparison between clustering-based under-sampling and random sampling with replacement.

3.2. Tomek Link Elimination

Tomek Link Elimination is a widely-applied conventional under-sampling technique. Unlike other under-sampling counterparts in the field, which will remove a large portion of the majority data to produce a nearly-balanced subset, Tomek Link will only remove the instances that overlapped with other items with different labels, which it determines as 'noise'. The idea for Tomek Link to find overlapped instances is to compute 'neighbour pairs', which means for two data x_i and x_j, if they are both the other's nearest neighbour in terms of some distance measures (say, Euclidean distance), then the pair of instances will be called 'Tomek link'. If the class labels of Tomek link are different, then we could remove the majority or the minority instance, or both. Formally, the algorithm could be described as:

For two instances x_i and x_j, if for any $x_k \in X \setminus \{x_i, x_j\}$, we could have $dist(x_i, x_j) < dist(x_i, x_k)$ and $dist(x_i, x_j) < dist(x_j, x_k)$, then x_i and x_j are called Tomek link. If the instances of a pair of Tomek link belongs to different classes, then we could remove one or both of them.

In our proposed algorithm, since the objective is to tackle imbalance data classification, we only remove the instance from the majority class with Tomek Link. The setting reflects our priority on the spotting the minority data, as this is usually the more important part in class-imbalanced data. And since we have multiple Xgboost classifiers, a sufficiently complicated decision boundary for the majority class could still be obtained. And regarding the distance measurement of Tomek Link, Euclidean distance could usually lead to satisfying performance, thus in our method this straightforward metric is implemented. Tomek Link Elimination is arranged after random under-sampling in our overall algorithm procedure, and each subset will be processed with this technique respectively. Notice that this procedure will result in a smaller size of the post-sampled majority class data, thus it justifies our setting of the p value in section 3.1. The effects of Tomek Link could be illustrated as Figure 2.

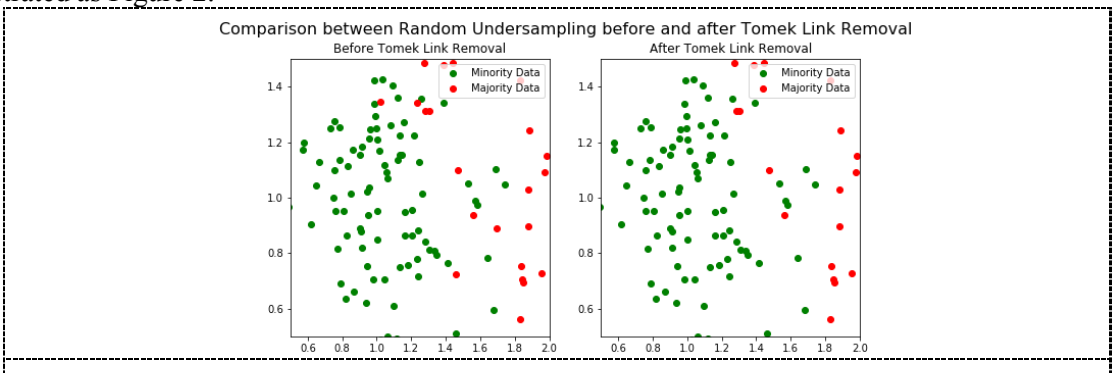

Figure 2. Demonstration of the effect of Tomek Link Elimination.

3.3. Xgboost

Xgboost is based on the algorithm of Gradient Boosting Machine (GBM), which is a popular Machine Learning technique firstly proposed in [22]. As a boosting ensemble method, it follows the idea to learn from the mistakes of the previous steps. Specifically, in GBM we use the gradient of the loss function with respect to the existed model to represent a 'pseudo-residual' between the existed predictions and the true labels/classes. Formally, at step K, suppose we already have a model of:

$$F_K(x) = \sum_{k=1}^{K} \alpha_k f_k(x; \theta_k) \tag{1}$$

Where $f_k(x; \theta_k)$ is the sub-model of the k-th step and α_k is the weight of the corresponding model. Then the 'pseudo-residual' of the $K+1$ step will be:

$$r_{K+1} = -\frac{\partial L(y, F_K(x))}{\partial F_K(x)} \tag{2}$$

Where $L(.,.)$ is the loss function that should be differentiable. Then the model of the $K+1$ step will be designed to fit this gradient as residual:

$$\theta_{K+1} = \arg\min_{\theta} L(r_{K+1}, f_{K+1}(x;\theta)) \tag{3}$$

And after obtaining the new additive model at the current step, we would like to add it back to the overall model $F_K(x)$ with a weighting parameter α obtained by line search:

$$\alpha_{K+1} = \arg\min_{\alpha} L(y, F_K(x) + \alpha f_{K+1}(x;\theta_{K+1})) \tag{4}$$

And finally, the holistic model at this step could be denoted by:

$$F_{K+1}(x) = F_K(x) + \alpha_{K+1} f_{K+1}(x;\theta_{K+1}) \tag{5}$$

And we could iteratively grow the model for given steps, or continue the additive fitting procedure until we get performances satisfying some specific metrics. However, when using the second paradigm in training, one should be aware of potential risks of overfitting. And in practice, like other Machine Learning algorithms, a regularisation term, typically l_2 norm of the parameter, will be added to the model to prevent overfitting.

3.4. Bagging of Xgboost and the Holistic Algorithm

As a bagging approach, in our algorithm each Xgboost model will be fitted with the post-Tomek Link under-sampled data respectively, and this will result in T separate classifiers, where T is the number of under-sampling subsets. Conventionally, for bagging of classifiers with two classes, the combining approach is to set the output of each sub-model $h(x) \in \{-1,1\}$ and sum the output labels, where the final output will follow the judgement of the majority. This kind of hard-threshold combination, however, will increase the opportunity of misclassification. A better approach would be to obtain a probability for instances to be classified to majority/minority, as we might want the instance to be easier to be determined as a majority/minority and more difficult for the opposite.

Our approach to get this probability is quite straightforward: since our topic is limited to two-class classification, we could get a weighted output with weights sum up to 1, and the final output will be the weights of the classifiers that predict the instance as 1 (minority class, in our settings). And this output could be naturally regarded as the probability of the instance to be in the minority class. Formally, this could be denoted as:

$$p(y_i = 1 | x_i) = \sum_{t=1}^{T} \alpha_t H_t(x_i) \tag{6}$$

In practice, we found that setting α values uniformly with $\alpha_t = 1/T$ could achieve a satisfying performance, and this could also be interpreted as the result of random sampling with replacement. With a probability output, we could use a sign function as follows to determine the class of an instance:

$$f(x_i) = \begin{cases} 1, & p(y_i = 1 | x_i) >= \delta \\ 0, & otherwise \end{cases} \tag{7}$$

And this setting enables us to tune the parameter δ with cross-validation and our emphasis on majority/ minority. A lower δ will lead to a higher recall of minority data, which could be sometimes crucial (say, the diagnosis of cancer, which a false negative will be life-threatening); and a higher δ value will lead to a higher recall of the majority data. The holistic procedure of the algorithm could be described as Figure 3.

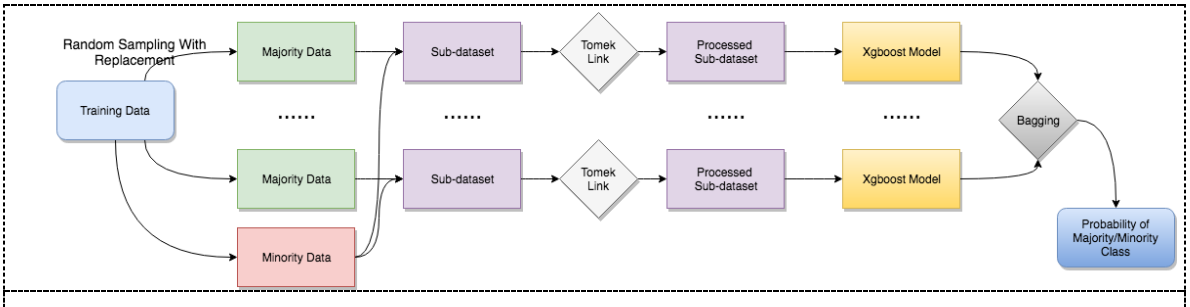

Figure 3. Process of the Holistic Algorithm

4. Experimental Results

In this section, we will be illustrating and discussing the experimental results of the proposed algorithm on a set of Mandarin radio broadcast data. Our task is to recognize if a keyword is presented in a record of radio. In this section, we will show that the proposed algorithm could considerably improve the performance of the recognition task under this dataset.

4.1. Data Composition and Preprocessing

The original data is composed of 133 records of radio broadcast, with few of them containing the keyword 'Beijing Time' (in Mandarin). After pre-processing them into 5-seconds short pieces and labelling the pieces containing the full record of keyword as 'keyword record', we obtained 6906 records in total, with 197 of them containing the keyword. Clearly, this is a label-imbalanced dataset with significant noise, thus it is an excellent example to test our algorithm. The original wave-signal type of data is rarely used in learning-based algorithms. Instead, following the conventional procedure, we transferred the records into MFCC features. And our goal is to build an algorithm to classify instances by their MFCC features.

4.2. Evaluation Metrics and Experiment

Since the classification of label-skewed data could achieve a high accuracy by simply predicting all the instance as majority, ordinary accuracy metric could not sufficiently represent the quality of algorithm. Alternatively, in label-imbalanced data classification, metrics of precision and recall will usually be employed. Using TP, FP, TN, FN to denote the classification results determined as True Positive, False Positive, True Negative and False Negative, then precision and recall of the positive class could be calculated by:

$$\begin{cases} precision = \dfrac{TP}{TP + FP} \\ recall = \dfrac{TP}{TP + FN} \end{cases} \tag{8}$$

And vice versa for the negative class. For specifically the positive or the negative class, we could compute the F1 score of the score:

$$F_1 = 2 * (\frac{precision * recall}{precision + recall}) \tag{9}$$

In our experiment, we focus on four evaluation metrics: overall classification accuracy, recall of the majority class, recall of the minority class, and the F1 score of the minority class. We adopt the F1 score of the minority class as the overall evaluation metric, as in our task is to spot the minority (keyword). And as discussed in section 3, the output of our model is the probability to classify the instance as minority (keyword, with label 1), and we could tune the value of δ to get the optimal prediction. In our experiments, the value of δ is tested from 0 to 1 (open interval) with a precision length of 0.05. The change of the classification accuracy, majority (non-keyword) recall and

minority(keyword) recall are collected as illustrated as figures. Moreover, the optimal behaviour of different types of algorithms are listed in a table.

There are 4 types of models tested in the experiments, with all of them based on Xgboost classifier and the parameters are best-tuned via validation. The benchmark model is a single Xgboost classifier with the full label-imbalanced data as the training set, and as we could see in figure 4, the problem of overfitting is of great significance. In the figure, the x-axis denotes the value of δ, and the y-axis represents the value of accuracy/recall as shown in the plots. Notice that for the single-classifier situation the output comes back to a hard-threshold decision problem, thus there are large flat-curve areas in the plots.

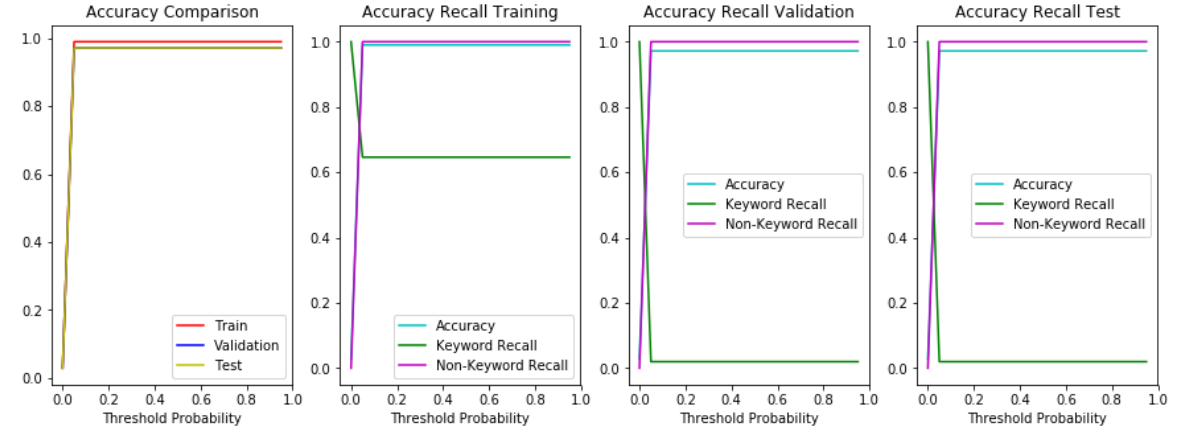

Figure 4. Performances of the benchmark single Xgboost with full dataset.

The second model we tested is the single Xgboost classifier with our under-sampling scheme. This method could be interpreted as 'single-model bagging', which could utilize the advantages of under-sampling and Tomek Link but does not enjoy the merits of bagging of classifiers. The performance of this kind of algorithm could be shown in figure 5. From the figure, it could found that the overfitting problem has been mitigated, and the classifier stop to predict most of the instances as majority. And although there is a decrease in the recall of the majority data, the overall performance has been improved.

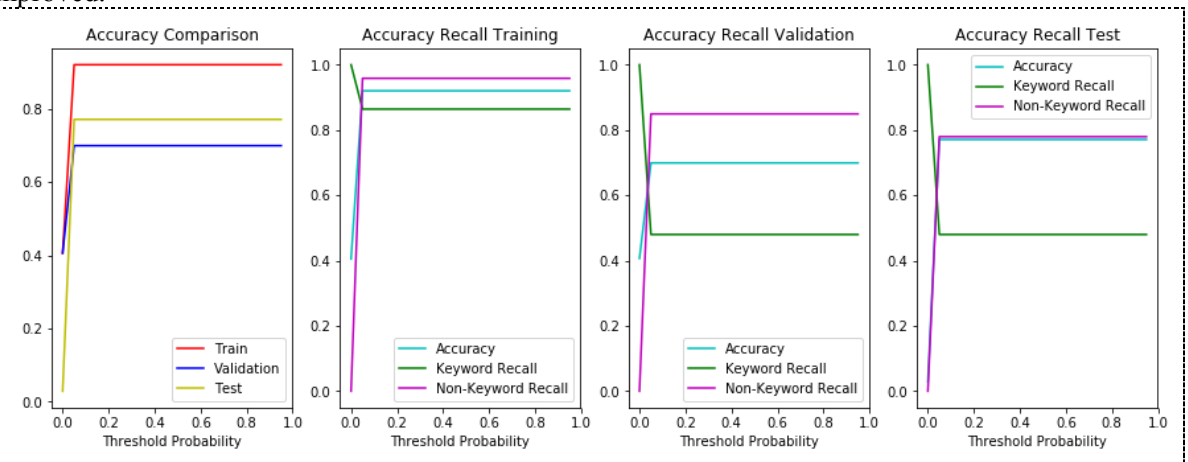

Figure 5. Performances of single Xgboost with the proposed under-sampling scheme.

And finally, the bagging of 5 and 10 Xgboost classifiers following the proposed paradigm are tested with the same dataset. The results of these two methods are illustrated in figure 6 and 7. Two major improvements could be found in the two figures: Firstly, the overall performance considering the hybrid of majority and minority recall has been significantly improved. We could find that the

values of the average of the recalls of majority and minority in Figure 6 and 7 could reach significantly higher levels. Secondly, as more than 1 classifiers are added to the holistic model, the differences between the varying values of δ begins to matter. And as we could see from the figure, the choice of δ could be accomplished via validation, and we could get the optimal output together with the prediction confidence of each instance.

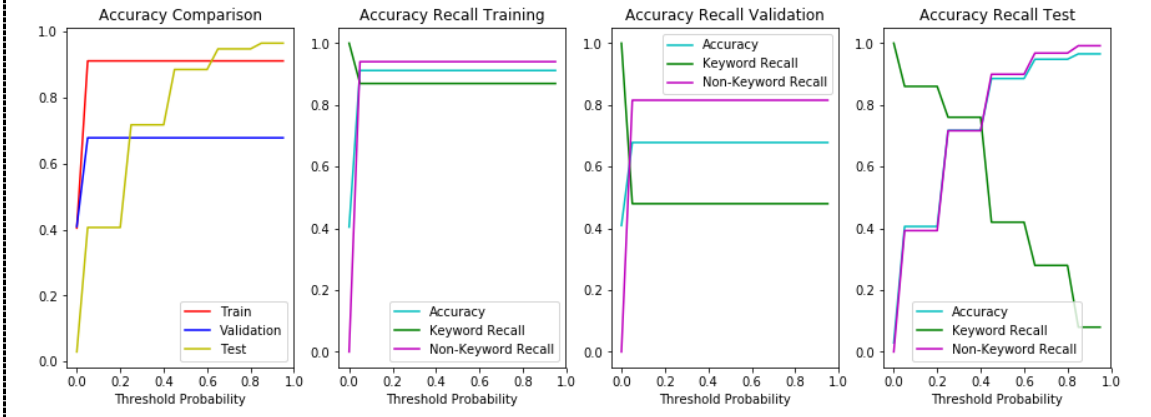

Figure 6. Performances of 5 Xgboost bagging with the proposed algorithm.

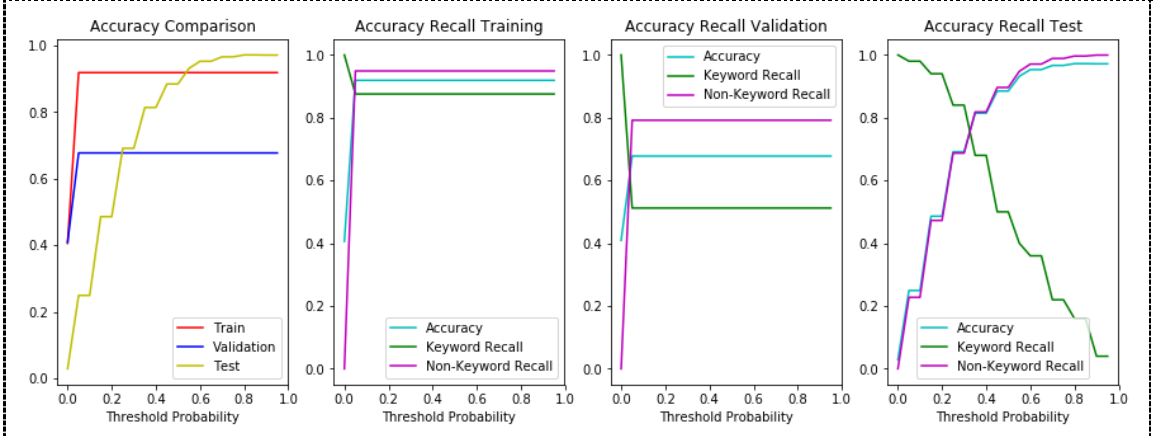

Figure 7. Performances of 10 Xgboost bagging with the proposed algorithm.

Table 1 shows the best F1 score of the test minority (keyword) data and the accuracy/ recall information at the level. There is an additional metric of 'balanced F1 score', which means to assume the minority and majority are in the same size to compute the F1 score. This metric could put additional emphasis on the minority data and increase the retrieved recall of minority (keyword).

4.3. Discussion

From the above experiments, we could found that the proposed bagging of Xgboost model could considerably improve the performance of label-imbalanced data classification. And in addition to accuracy improvement, there also exist other advantages from different aspects in our algorithm, which worth further discussions.

Table 1. Performance Comparison between Different Models

	Best F1 Score	Balanced F1 Score	Minority Recall	Majority Recall	Accuracy	δ
Benchmark	0.0392	0.0392	0.02	**1.0**	**0.9716**	-
1 Xgboost	0.1081	0.5645	0.48	0.7795	0.7708	-

5 bagging Xgboost	0.1300	-	0.32	0.8927	0.8762	0.45
	-	0.6871	0.68	0.7008	0.7002	0.25
10 bagging Xgboost	**0.3077**	-	0.36	**0.9708**	**0.9531**	0.6
	-	**0.7803**	**0.84**	0.6871	0.6915	0.25

The first remark is the preferable time complexity property of our algorithm. With the under-sampling procedure, the sample size of each training set for individual classifiers decreases, which brings the bonus advantage of less demanded training time. Notice that as a boosting method, the training time of Xgboost will considerably raise when the size of the training set is large. In our experiments, the benchmark classifier, which was trained on the full dataset and achieved the worst performance, actually takes the longest time to train. And for the bagging model, the time complexity should be $O(K \bullet M)$, where K is the number of classifiers and M is the time complexity of individual classifier. This is a favourable property, as we could efficiently fit a model with linear time complexity.

The second point to argue is the probabilistic output, which could lead to benefits more than the simple 'soft output' and 'threshold flexibility'. Probabilistic output could reflect the level of confidence, which could be of great importance under some circumstance. The probabilistic output could also bring benefits in evaluation, as we could compute the cross-entropy loss between the prediction and the true labels. And in the two-class scenario, the log-likelihood is equivalent to the negative cross-entropy, thus we could possibly perform model selection with metrics like AIC and/or BIC.

5. Conclusion

In this paper, a novel algorithm based on bagging of Xgboost classifiers is proposed for label-imbalanced noisy data classification. The bagging procedure is designed with random under-sampling with replacement and Tomek Link elimination to generate data and prevent noise, and each classifier is trained independently with the specific subset of data. After obtaining multiple models, we could combine them with uniformly-distributed weights and transform the output into the probability for the instances to be the majority/minority. The performance of the method has been tested on Mandarin radio broadcast data for keyword recognition task, and experimental results have shown that the proposed algorithm could outstrip simple Xgboost, and adding amounts of bagging groups could improve the performance.

To this end, the paper made the following major contributions. Firstly, it designed a novel algorithm to tackle the classification task for label-imbalanced data with high level of noise. The designed model could outperform existed models and the efficiency is in a high grade. Secondly, the paper applied Xgboost, a recently-proposed GBM-based toolkit, to the problem of imbalance data classification and explored its capability in dealing with such problems. The innovative work could provide copious information for further research with the same technique. And finally, the paper elaborated on the idea to use under-sampling and bagging for imbalanced data classification and made major contribution to the development of this branch of algorithms.

Acknowledgments

The authors would like to thank the assistances and encouragements from colleagues, and the support from National Science Foundation of China (Grant: NSFC51475391), University-Enterprise Cooperation Project (17H1199).

Reference

[1] Russell, S J and Norvig P 2016 *Artificial intelligence: a modern approach* (Malaysia: Pearson Education Limited)

[2] Krawczyk B 2016 Learning from imbalanced data: open challenges and future directions *Progress in Artificial Intelligence* pp 221-232

[3] Ali A, Siti M S and Anca L R 2015 Classification with class imbalance problem: a review *Int. J. Advance Soft Compu. Appl*
[4] Qiong G, Cai Z, Zhu L and Huang B 2008 Data mining on imbalanced data sets *In Advanced Computer Theory and Engineering. ICACTE'08. International Conference on Soft* (IEEE:2008) pp. 1020-1024
[5] Chen T and Carlos G 2016 Xgboost: A scalable tree boosting system *Proceedings of the 22nd acm sigkdd international conference on knowledge discovery and data mining* (ACM:2016)
[6] Branco P, Luís T and Rita P R 2016 A survey of predictive modeling on imbalanced domains *ACM Computing Surveys (CSUR)* p 31
[7] Burnaev E, Erofeev P and Papanov 2015 A Influence of resampling on accuracy of imbalanced classification *In: Eighth International Conference on Machine Vision (ICMV 2015), volume 9875* (International Society for Optics and Photonics) p. 987521.
[8] Weiss G M, McCarthy K and Zabar B 2007 Cost-sensitive learning vs. sampling: Which is best for handling unbalanced classes with unequal error costs? *DMIN* pp.35-41
[9] Mani I and Zhang I 2003 KNN approach to unbalanced data distributions: a case study involving information extraction. *In Proceedings of workshop on learning from imbalanced datasets*
[10] Hart P 2006 The condensed nearest neighbor rule (corresp.) *IEEE Trans. Inf. Theor. (vol 14)*. pp 515–516
[11] Lemaître G, Fernando N and Christos K A 2017 Imbalanced-learn: A python toolbox to tackle the curse of imbalanced datasets in machine learning *Journal of Machine Learning Research 18.17* pp. 1-5
[12] Yen S and Lee Y 2009 Cluster-based under-sampling approaches for imbalanced data distributions *Expert Systems with Applications* pp 5718-5727
[13] Sobhani P, Viktor, H and Matwin S 2014 Learning from imbalanced data using ensemble methods and cluster-based undersampling *In International Workshop on New Frontiers in Mining Complex Patterns* (Springer, Cham.) pp. 69-83
[14] Woźniak M, Graña M and Corchado E 2014 A survey of multiple classifier systems as hybrid systems *Information Fusion* pp. 3-17
[15] Galar M, Fernandez A, Barrenechea E, Bustince H and Herrera F 2012 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)* pp. 463-484
[16] Wang Q, Luo Z, Huang J, Feng Y and Liu Z 2017 A novel ensemble method for imbalanced data learning: bagging of extrapolation-SMOTE SVM. *Computational intelligence and neuroscience*
[17] Omar KBA 2018 XGBoost and LGBM for Porto Seguro's Kaggle challenge: A comparison *Preprint Semester Project* pub.tik.ee.ethz.ch
[18] Song Y 2018 Stock Trend Prediction: Based on Machine Learning Methods *Master's thesis, UCLA*
[19] Torlay L, Perrone-Bertolotti M, Thomas E and Baciu M 2017 Machine learning–XGBoost analysis of language networks to classify patients with epilepsy *Brain informatics* pp. 159-169
[20] Wang C, Wang S, Shi F and Wang Z 2018 Robust Propensity Score Computation Method based on Machine Learning with Label-corrupted Data *preprint* arXiv:1801.03132
[21] Bühlmann P 2012 Bagging, Boosting and Ensemble Methods *Handbook of Computational Statistics* (Springer: Berlin, Heidelberg). pp. 985-1022.
[22] Friedman J H 2001 Greedy function approximation: a gradient boosting machine *Annals of statistics* pp. 1189-1232