An Improved Adaboost Algorithm for Imbalanced Data Based on Weighted KNN

Kewen Li
China University of petroleum
College of Computer & Communication Engineering
Qingdao, Shandong Province, China
e-mail: likw@upc.edu.cn

Peng Xie
China University of petroleum
College of Computer & Communication Engineering
Qingdao, Shandong Province, China
e-mail: upc_xp@126.com

Jiannan Zhai
Florida Atlantic University
Institute for Sensing and Embedded Network Systems Engineering
777 Glades Road, Boca Raton, FL, USA
e-mail: jzhai@fau.edu

Wenying Liu
China University of petroleum
College of Computer & Communication Engineering
Qingdao, Shandong Province, China
e-mail: wyliu@upc.edu.cn

***Abstract*—Imbalanced data become an obstacle in data mining nowadays, minority class sometimes are more important than majority class, just like in medical diagnosis, credit card fraud and etc. This paper focuses on the imbalanced data problem that adaboost algorithm cannot get a proper accuracy rate for minority class, and propose an improved adaboost algorithm for imbalanced data based on weighted KNN(K-Adaboost). K-Adaboost uses KNN algorithm to cut down majority class weights which is near to minority class, so that the classify can pay more attention to minority class. Besides, the paper uses a new error function and sets a threshold during classifying process in order to avoid weight distortion.**

Keywords—Adaboost; k-NearestNeighbor; imbalanced data; classification

I. INTRODUCTION

Classification is one of the important research contents in the field of data mining, some of the classification method has been relatively mature, which perform well in classifying balanced data. Now almost classifiers are designed on the hypothesis: class distribution are roughly balanced. However, this hypothesis is not established, the number of samples of a class in the data set may be far less than the other categories.

Imbalanced data exists in many areas, just like information retrieval, credit card fraud, medical diagnosis and so on, minority class is more important than majority class in these areas. If a healthy person is misdiagnosed as patient in medical diagnosis, it will bring him a burden and a bad influence on doctor; If a patient is diagnosed as healthy person, then it may miss the best treatment period, which will cause serious consequences. Traditional classification methods have a high accuracy rate for majority class, but a low accuracy rate for minority class[1]. So the classification of imbalanced data need new methods[2].

Adaboost[3] algorithm is a universal learning algorithm, can be directly used to classify the imbalanced data. It can be regarded as a gradient descent algorithm in the function space[4], but samples of majority class are far more than samples of minority class, so the traditional adaboost algorithm will pay more attention to error samples which is classified by existing classifiers, the classifier will not perform well in classify minority class. According to this situation, Viola and Jones[5] proposed AsymBoost algorithm, for the error of majority class and minority class, an increase of different weights, small samples increased weight greater, so that a classifier will perform better on minority class. Wu yan[6] add the training weight factor, the weight of the positive sample is dynamically controlled, and the classification performance and convergence rate are improved. The traditional AdaBoost algorithm may face a problem of the weight distribution distortion, which makes the classifier over fitting to one class, and the existence of bias to other classes. Li[7] adjusts the weights of the samples in each class inside to deal with this problem.

The paper proposes K-Adaboost. The error function is replaced by the product of the majority class accuracy and the minority class accuracy in order to reflect the performance of the classifier in the imbalanced data, then the weight function is replaced by Sigmoid function, so the system will give higher weight to the classifier which performs well. Besides, The algorithm uses k-NearestNeighbor(KNN) algorithm to cut down majority class weights which is near to minority class, so that the classify can pay more attention to minority class.

II. RELATED WORK

A. Imbalanced Data

Any data set that exhibits an unequal distribution between its classes can be considered imbalanced[8]. However, the common understanding in the community is that imbalanced data correspond to data sets exhibiting significant, and in some cases extreme, imbalances. As for binary classification problem, assuming that our data set is S, the majority class of the data set is called S_maj, and the minority class of the data set is called S_min, usually the Ratio of S_maj to S_min may be 100:1, 1000:1 or even 10000:1. The study of imbalanced data is to learn the useful information in the data set that is not uniform.

978-1-5090-3619-6/17/$31.00 ©2017 IEEE

At present, the methods to deal with the class imbalance data can be divided into two classes [9,10]. One is the method of processing samples before training the classifier, the most commonly used is the sampling technology. The basic idea is to eliminate or reduce the imbalance of the data by changing the class distribution of the training data, which can be divided into undersampling and oversampling [11,12]. The other is to improve the learning algorithm which can be applied to imbalanced data classification, the most common method is cost sensitive learning algorithm[13,14] and Boosting based method[15,16].

B. AdaBoost

AdaBoost[17] is a typical learning algorithm, which can effectively improve the ability of classifier by several iterations. In the initialization, all the training samples are assigned to the same weight, and then we can get several weak classifiers by several rounds of training. After the end of each round of training, calculating error rate of the weak classifier, improving the weights of samples which is classified wrong, and reducing the weights of samples which is classified correctly. Finally, these weak classifiers become a strong classifier to complete the classification task.

Adabbost algorithm can be regarded as a gradient descent algorithm in the function space[18], where the cost function is minimized by gradient descent algorithm. In each iteration, we will choose new weak classifier whose direction is near to negative gradient direction into strong classifier.

The target of adaboost algorithm to optimize the overall accuracy. The error caused by misclassification of different classes is same. Adaboost algorithm will pay more attention to error samples which is classified by existing classifiers, the classifier will not perform well in classify minority class. It is clear that this is not a very good learning effect, so the traditional learning algorithm has great limitations in the imbalanced data learning.

AdaBoost algorithm:

Input : Dataset $T = \{(x_1, y_1), (x_2, y_2), \cdots, (x_N, y_N)\}$, $x_i \in R^n$, $y \in Y = \{-1, +1\}$

Output : Strong classifier G(x)。

(1) Initialization

$$D_1 = (w_{11}, \cdots, w_{1i}, \cdots, w_{1N}), w_{1i} = \frac{1}{N}, i = 1,2, \cdots, N \quad (1)$$

(2) For $m = 1,2, \cdots, M$

(a)Get weak classifier by weight distribution D_m

$$G_m(x) = \{-1, +1\} \quad (2)$$

(b)Compute the error on dataset by $G_m(x)$

$$e_m = P(G_m(x) \neq y_i) = \sum_{i=1}^{N} w_{mi} I(G_m(x) \neq y_i) \quad (3)$$

(c)Compute the weight of $G_m(x)$

$$\alpha_m = \frac{1}{2} log \frac{1-e_m}{e_m} \quad (4)$$

(d)Update D_m

$$D_{m+1} = \left(w_{m+1,1}, \cdots, w_{m+1,i}, \cdots, w_{m+1,N}\right) \quad (5)$$

$$w_{m+1,i} = \frac{w_{mi}}{Z_m} \exp(-\alpha_m y_i G_m(x_i)) \quad (6)$$

$$Z_m = \sum_{i=1}^{N} w_{mi} \exp(-\alpha_m y_i G_m(x_i)) \quad (7)$$

(3) Get the strong classifier

$$f(x) = \sum_{m=1}^{M} \alpha_m G_m(x) \quad (8)$$

$$G(x) = \text{sign}\left(f(x)\right) = \text{sign}\left(\sum_{m=1}^{M} \alpha_m G_m(x)\right) \quad (9)$$

III. K-ADABOOST ALGORITHM

A. K-AdaBoost Algorithm

The paper proposes K-Adaboost algorithm. The error function is replaced by the product of the majority class accuracy and the minority class accuracy in order to reflect the performance of the classifier in the imbalanced data, then the weight function is replaced by Sigmoid function, so the system will give higher weight to the classifier which performs well. Besides, The K-Adaboost algorithm uses KNN algorithm[19] to cut down majority class weights which is near to minority class, so that the classify can pay more attention to minority class.

K-AdaBoost algorithm:

Input : Dataset $T = \{(x_1, y_1), (x_2, y_2), \cdots, (x_N, y_N)\}$, $x_i \in R^n$, $y \in Y = \{-1, +1\}$

Output: Strong classifier G(x)

(1) Initialization

$$D_1 = (w_{11}, \cdots, w_{1i}, \cdots, w_{1N}), w_{1i} = \frac{1}{N}, i = 1,2, \cdots, N \quad (10)$$

(2)For $m = 1,2, \cdots, M$

(a) Get weak classifier by weight distribution D_m

$$G_m(x) = \{-1, +1\} \quad (11)$$

(b) Compute the new error on dataset by $G_m(x)$

$$e_m = P(G_m(x) \neq y_i \;\&\&\; y_i = -1) \cdot P(G_m(x) \neq y_i \;\&\&\; y_i = +1) = (\sum_{i=1}^{N} w_{mi} I(G_m(x) \neq y_i \;\&\&\; y_i = -1)) \cdot (\sum_{i=1}^{N} w_{mi} I(G_m(x) \neq y_i \;\&\&\; y_i = +1)) \quad (12)$$

(c) Compute the weight of $G_m(x)$ by sigmoid function

$$\alpha_m = \frac{1}{2} \cdot \frac{1}{1+e^{-(1-e_m)}} \quad (13)$$

(d) Update D_m

$$D_{m+1} = \left(w_{m+1,1}, \cdots, w_{m+1,i}, \cdots, w_{m+1,N}\right) \quad (14)$$

$$w_{m+1,i} = \frac{w_{mi}}{Z_m} \exp(-\alpha_m y_i G_m(x_i)) \quad (15)$$

$$Z_m = \sum_{i=1}^{N} w_{mi} \exp(-\alpha_m y_i G_m(x_i)) \quad (16)$$

(e) Repeat updating D_m by KNN

1. Set threshold ρ, find significant misclassification of minority class

$$\mathrm{L} = (l_1, l_2, \cdots, l_k) =$$

$$D_m(w_{mi} > \rho \;\&\&\; y_i = -1 \;\&\&\; G_m(x_i) = +1) \quad (17)$$

2. Use KNN in $\mathrm{L}(1,2,\cdots,\mathrm{k})$, find $\mathrm{L}' = (l_{11}, \cdots, l_{k\ldots})$

3. Find Samples that belong to the majority class and are classified correctly in L', then reduce these samples' weight

$$\mathrm{IF}\left(x_i \in l_{kp} \;\&\&\; x_i \in S_{max} \;\&\&\; G_m(x_i) = +1 \right)$$

$$w_{m,i} = \max\{\frac{w_{m,i}}{k_i}, \frac{1}{\mathrm{m}}\} \quad (18)$$

$$Z_m = \sum_{i=1}^{N} w_{mi} \quad (19)$$

$$w_{m+1,i} = \frac{w_{mi}}{Z_m} \quad (20)$$

$$D_{m+1} = \left(w_{m+1,1}, \cdots, w_{m+1,i}, \cdots, w_{m+1,N}\right) \quad (21)$$

(3) Get the strong classifier

$$\mathrm{f(x)} = \sum_{m=1}^{M} \alpha_m G_m(x) \quad (22)$$

$$\mathrm{G(x)} = \mathrm{sign}\left(\mathrm{f(x)}\right) = \mathrm{sign}\left(\sum_{m=1}^{M} \alpha_m G_m(x)\right) \quad (23)$$

The error function (12) of K-AdaBoost is different from AdaBoost. The error function is replaced by the product of the majority class accuracy and the minority class accuracy in order to reflect the performance of the classifier in the imbalanced data.

K-Adaboost algorithm uses KNN algorithm to cut down majority class weights which is near to minority class (17-20). Firstly, choosing minority class samples L which is wrong classified (17). Then choosing majority class samples L' that is near to each samples in L by KNN. If l_{kp} in L' is correctly classified, cut $l_{kp}{}'s$ weight (18). k_i is a count, if x_i belongs to majority class and it is classified correctly, then $k_i = k_i + 1$.

B. Evaluation

Considering a basic two-class classification problem, let {p,n} be the true positive and negative class label and {Y,N} be the predicted positive and negative class labels. Then, a representation of classification performance can be formulated by a confusion matrix (contingency table), as illustrated in Table 1.

Accuracy= $\frac{(\mathrm{TP+TN})}{(\mathrm{TP+TN+FP+FN})}$ is a common evaluation criteria in classification, It reflects the performance of the classifier on data sets, But not accurately reflect the performance of classifier on imbalanced data sets. According to this situation, we should use more reasonable evaluation criteria. AUC[20] (Area Under Curve) is equivalent to the probability that a randomly chosen positive example is ranked higher than a randomly chosen negative example, it is the area of the curve ROC (Receiver Operating Characteristic)[21].

$$\mathrm{TNP=TP/\ ((FP+FN))} \quad (24)$$

$$\mathrm{TNR=TN/((TN+FP))} \quad (25)$$

$$\mathrm{FPR=FP/(TN+FP)} \quad (26)$$

$$\mathrm{FNR=FN/(TP+FN)} \quad (27)$$

$$\mathrm{AUC=(1+TPR-FPR)/2} \quad (28)$$

The performance of the classifier is a positive proportion of the value of AUC. The higher the value of AUC, the better the performance of the classifier. AUC can be computed by the above calculation formula (24-28).

TABLE I. CONFUSION MATRIX

	p	n
Y	TP (true positive)	FP (false positive)
N	FN (false negative)	TN (true negative)

IV. SIMULATION

The paper test the proposed algorithm on the horseColic data set form UCI repository and KC1 from NASA data set. And weak classifier is generated by DecisionStump in the simulation. And the AUC of K-AdaBoost and AdaBoost is shown in Fig. 1-Fig. 4

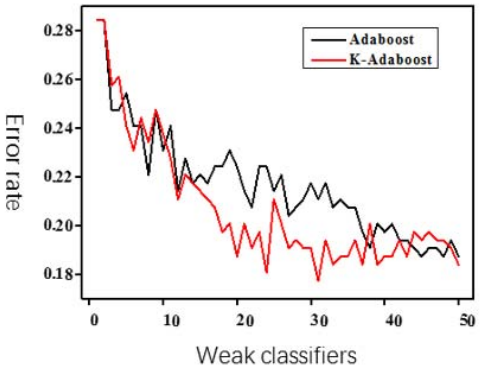

Figure 1. Error rate on horseColic

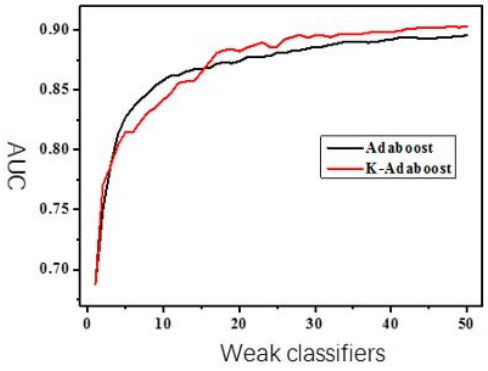

Figure 2. AUC on horseColic

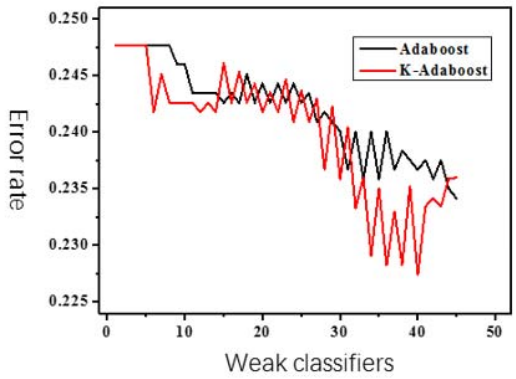

Figure 3. Error rate on PC1

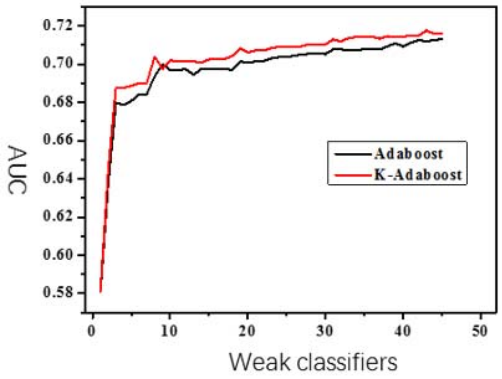

Figure 4. AUC on PC1

We can see that the classification error rate and AUC value from the experiment results on horseColic and KC1 data set. We can get several conclusions as follows:

The error rate of K-AdaBoost is faster than AdaBoost to reach a lower state. Because K-AdaBoost sets a minimum weight threshold and a strategy to cut down high weights, it can avoid weight distortion caused by excessive weights.

The performance of K-AdaBoost is significantly improved compared with the traditional AdaBoost algorithm. Because imbalanced data is taken into consideration in K-AdaBoost, AUC and error rate can get further optimization.

V. CONCLUSION

Aiming at the problem of imbalanced data, this paper proposes K-AdaBoost algorithm. The error function is replaced by the product of the majority class accuracy and the minority class accuracy in order to reflect the performance of the classifier in the imbalanced data, then the weight function is replaced by Sigmoid function. Besides, The K-Adaboost algorithm uses KNN algorithm to cut down majority class weights which is near to minority class, so that the classify can pay more attention to minority class. The system will give higher weight to the classifier which performs well and avoid weights distortion.

The K-AdaBoost' efficiency is better than AdaBoost algorithm on imbalanced data, whose Error rate and AUC value is improved. This research is a meaningful attempt to help deal with imbalanced data, and we will focus on objective function to deal with imbalanced data in further research.

ACKNOWLEDGMENTS

The authors acknowledge the support of the Natural Science Foundation of Shandong Province, China (No.ZR2013FL034).

REFERENCE

[1] Weiss GM, Provost F. Learning when training data arecostly: theeffect of class distribution on tree induction .Journal of Artificial Intelligence Research, 2003, 19:315-354.

[2] Zadronzny B, Elkan C. Learning and making decisions when costs and probabilities are both unknown. Proceedings of the 7th International Conference on Knowledge Discovery and Data Mining. New York, USA:ACM, 2001:204-213.

[3] Freund Y, Schapire R E.A decision - theoretic generalization of on-line learning and an application to boosting[J].Journal of Computer and System Sciences, 1997, 55(1) : 119- 139.

[4] JIANG Yan, DING Xiaoqing, AdaBoost algorithm using multi-step correction [J]. Tsinghua Univ (Sci& Tech) , 2008 , 48(10):1613-1616.

[5] Viola P, Jones M.Fast and robust classification using asymmetric AdaBoost and a detector cascade[J]. Advances in Neural Information Processing Systems, 2002, 14: 1311- 1318.

[6] WU Yan, XIANG Enning, Dynamic Weights and Pre-partitioning Real-Adaboost Face Detection Algorithm[J], Compute Engineering, 2007, 33(3):208-209.

[7] Li Bin , Research on classifier optimization algorithm [D] , computer software and theory, 2003.

[8] Haibo He, Member, IEEE, and Edwardo A. Garcia. Learning from Imbalanced Data[J]. IEEE TRANSACTIONS ON KNOWLEDGE AND DATA ENGINEERING, VOL. 21, NO. 9, SEPTEMBER 2009.

[9] Weiss G.Mining with rarity: a unifying framework[J]. SIGKDD Exploration,2004, 6(1) : 7- 19.

[10] Kotsiantis S, Kanellopoulos D, Pintelas P.Handling imbalanceddatasets: a review[J]. GESTS International Transactions on ComputerScience and Engineering, 2006, 30.

[11] Batista G E, Prati R C, Monard M C.A study of the behavior ofseveral methods for balancing machine learning training data [J].ACM SIGKDD Explorations Newsletter, 2004, 6(1) : 20- 29.

[12] Van Hulse J, Khoshgoftaar T M, Napolitano A.Experimental perspectiveson learning from imbalanced data[C]//Proceedings of the 24thinternational conference on Machine learning, 2007, 227: 935-942.

[13] Maloof M A, Langley P, Binford T O, et al.Improved rooftop detectionin aerial images with machine learning[J].Machine Learning,2003, 53: 157- 191.

[14] Huang Kaizhu, Yang Haiqin, King Irwin, et al.Learning classifiersfrom imbalanced data based on biased minimax probability ma -chine[C] //Proceedings of the IEEE Computer Society Conference on Computer Vision and Pattern Recognition, 2004, 2: 558- 563.

[15] Viola P, Jones M.Fast and robust classification using asymmetricAdaBoost and a detector cascade[J]. Advances in Neural InformationProcessing Systems, 2002, 14: 1311- 1318.

[16] Karakoulas G, Shawe - Taylor J.Optimizing classifiers for imbal -anced training sets[C] //Proceedings of the 1998 Conference on Advancesin Neural Information Processing Systems, 1999, 11: 253-259.

[17] Chawla N V, Lazarevic A, Hall L O, et al.SMOTEBoost: improving prediction of the minority class in boosting[C]//Proceedings of the 7th European Conference on Principles and Practice of Knowledge Discovery in Databases, 2003: 107- 109.

[18] Joshi M, Kumar V, Agarwal R.Evaluating boosting algorithms to classify rare classes: Comparison and improvements [C]//Proceedings of the 1st IEEE International Conference on Data Mining, 2001: 257-264.

[19] JI Chengheng, LEI Yongmei. KNN-based even sampling preprocessing algorithm for big dataset [J]. JOURNAL OF SHANGHAI UNIVERSITY (NATURAL SCIENCE). Vol. 22 No. 1. Feb. 2016, 1007-2861(2016)01-0028-08.

[20] Hand D J. Measuring classifier performance: a coherent alternative to the area under the ROC curve[J]. Machine learning, 2009, 77(1): 103-123.

[21] Buying PMO R, Matter WV. The Use of the Area Under the ROC Curve in the Evaluation of Machine Learning Algorithms[J]. 1997, 30(7): 1145-1159.