Solving the Data Imbalance Problem of P300 Detection via Random Under-Sampling Bagging SVMs

Xiaofeng Shi, Guoqiang Xu, Furao Shen*, Jinxi Zhao
the National Key Laboratory for Novel Software Technology
Department of Computer Science and Technology, Nanjing University, China
Email: cell000feng@gmail.com

***Abstract*—The imbalance problem exists in P300 EEG data sets because P300 potential are collected under the condition of Oddball experimental paradigm. Hence, a P300 detection method, namely RUSBagging SVMs, is proposed in this paper to solve the imbalance problem and make an improvement. This algorithm re-samples the data sets at first to generate a rebalanced training set in one round of iteration and trains an SVM classifier based on the training set. Next, the SVM classifiers are integrated to make a final decision. In the integration of several classifiers, the information that is lost in the under-sampling process is generally considered. Therefore, the method is relatively robust. The experiments of character recognition based on P300 EEG data signals are conducted to examine the method. It is concluded from the experiments that RUSBagging method can indeed improve the performance of P300 detection by solving the imbalance problem in EEG data sets.**

I. Introduction

The brain of human beings controls the entire body by its electrical activity. If stimulation is given to the brain, Event Related Potential (ERP) comes into being. Among so many kinds of ERP Electroencephalography (EEG) Signals P300 has been successfully used in several BRAIN-COMPUTER INTERFACE (BCI) systems. The P300 potential occurs approximately 300ms after the subject is exposed to a certain stimuli. According to this feature, we can find out if an incident contains a certain stimuli by the recognition of P300 potential from the EEG signals. During years of study of the P300 recognition technology, a number of recognition methods have been proposed including linear or nonlinear classifiers like Neural networks[1][2], Support Vector Machines (SVMs)[3] for classification and other strong feature extraction techniques such as Independent Component Analysis (ICA)[4], Wiener Filter[5], and Continuous Wavelet Transform (CWT)[6].

Many studies show that the class imbalance problem of the data set may have influence to its classification[7]. However, the class imbalance problem has long been existed in P300 data set as a result of Oddball experimental paradigm[8], but hasn't attached much attention to researchers. This paper studies the imbalance problem of P300 data sets and shows that by reducing the imbalance ratio of the data sets we can improve the performance of the classifiers in P300 detection. Moreover, in accordance with the imbalance problem and the features of P300 EEG data sets, we proposed an integration method of SVMs, namely RUSBagging SVMs, which is shown to be effective in P300 detection.

At present, there are two main approaches to solve the imbalance problem[9]. One way is in the data-level and changing the training data distribution. Thus we can get the re-balanced data sets and then train classifiers on the new balanced or lower imbalanced ratio data sets. The other way is in the algorithm-level and strengthens the existing classifiers by adjusting algorithms to pay more attention to the smaller classes.

In this paper we focus to solve the imbalance problem of the P300 data sets in the data-level. Under-sampling is one of the major methods. It samples both the bigger and smaller class one by one so that we can get two new subsets with the same size. However, the instances that are not sampled are useless. It means some important points of the original sets may be lost, which is the main disadvantage of the under-sampling method. To respond to this problem, we decide to under-sample the original sets for several times to cover as many instances as possible. Then the SVM classifiers are trained based on these re-sampled data sets. To make a final decision, all the classifiers are integrated. In this way, the training set in every round of iteration is balanced. Then the experiments of character recognition based on P300 EEG signals are conducted to examine our method. It can be proved from the experiments that our method is quite effective for P300 detection, and improvements can be made if we properly deal with the imbalance problem of the data sets.

II. Imbalance problem in P300 Data

Why do the P300 EEG data sets have the imbalance problem? To answer the question, let's see how the P300 potential is generated firstly. To evoke the P300 potential, the P300 speller diagram proposed by Farwell and Donchin[10] is used. As it is shown in Fig. 1, the P300 speller diagram is composed of a 6 by 6 matrix of 36 characters. In the EEG collection experiment, the subject is asked to focus on the target character and count how many times the character has been intensified during one experiment trial. Within this trial, each of the rows and columns of the matrix is intensified by a random sequence, but only two of these 12 rows and columns contains the target character and can evoke P300 potential. Therefore, if we figure out which row and column contains the P300 potential from the 12 instances of one trial, we can know

Furao Shen is the corrresponding author.

978-1-4799-1959-8/15/$31.00 ©2015 IEEE

Fig. 1. Stimulus matrix of the P300 speller diagram

the target character. So the EEG data instances that contain P300 potential are labeled as positive class, while those don't are labeled as negative class.

If the data samples collected in one trial are described as

$$D = \{(x_1, y_1), (x_2, y_2) ..., (x_{12}, y_{12})\}, y_i \in \{-1, 1\}$$

then we can figure out that the ratio of negative class and positive class is

$$\frac{\sum_{i=1}^{12} I(y_i = -1)}{\sum_{i=1}^{12} I(y_i = 1)} = 5:1 \tag{1}$$

In the practical experiment, the subject is asked to spell a number of characters, which means the ratio of negative class and positive class of the total data sets is also 5:1.

Furthermore, the reason why the P300 EEG data sets have the imbalance problem comes from how the P300 potential comes into being. Among different ways to evoke P300 potential, visual evoking is more stable than touch evoking and auditory evoking and becomes the most common way. P300 speller diagram is one of the most popular visual evoking methods. Beside this, many other improved methods have been proposed by researchers. Whatever method is used to evoke P300 potential, the process must obey oddball experimental paradigm, which requests that the frequency of target stimulus to evoke P300 potential is smaller than non-target stimulus. As it has been shown in clinical studies, the amplitude of P300 potential has negative correlation with the frequency of target stimulus. As a result, when conducting an experiment to evoke P300 potential, it is necessary to keep the frequency of target stimulus in a small value. In the P300 speller diagram, for example, the frequency of target stimulus is $\frac{1}{6}$.

Assuming that in an evoke experiment of P300 potential, the frequency of target stimulus is $Freq_1$, and the frequency of non-target stimulus is $Freq_2$, according to oddball paradigm, we know

$$Freq_1 < Freq_2 \tag{2}$$

The data set in one trial we get from the experiment is

$$D = \{(x_1, y_1), (x_2, y_2) ..., (x_N, y_N)\}, y_i \in \{-1, 1\}$$

where N is the total number of stimulus. In one trial, every kind of stimulus appears for one time and only the target stimulus can evoke P300 potential. So we know

$$\frac{\sum_{i=1}^{N} I(y_i = -1)}{\sum_{i=1}^{N} I(y_i = 1)} = \frac{Freq_1}{Freq_2} < 1 \tag{3}$$

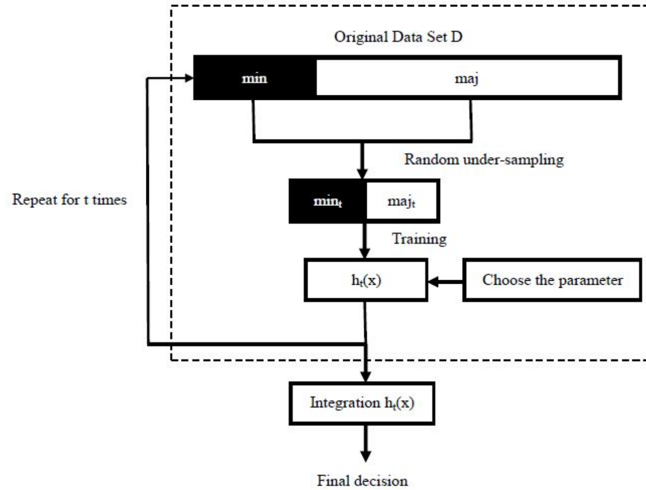

Fig. 2. The brief process of RUSBagging SVMs

Formula(3.3) shows that the class imbalance problem generally exists in P300 EEG data sets.

From the analysis, we can conclude that the P300 EEG data sets are imbalance data sets naturally, and the imbalance ratio (IR) of P300 speller diagram is 5:1. However, through so many years of study of the P300 EEG detection, the imbalance problem in the data sets hasn't attracted researchers' much attention. In this point, it is of great importance to have a clear understanding of imbalance problem in P300 data sets and find proper solutions.

III. Proposed Method: Random Under-Sampling Bagging SVMs

In the 3rd Wadsworth BCI competition of 2005, Alain Rakotomamonjy[3] won the first prize by proposing the method Ensemble SVMs. He divides the training set into 17 groups and trains 17 SVMs based on the 17 groups of data. Then the 17 SVMs are integrated by cross validation to make a final decision. However, the algorithm of Ensemble SVMs doesn't take account of the imbalance problem. The data set of every training group is still imbalanced, and integration doesn't contribute to solving this.

Inspired by Rakotomamonjy's[3] method, we proposed a classification algorithm to effectively solve the imbalance problem of the data sets by under-sampling and then use the ensemble learning method to avoid the loss of important data points when sampling.

Fig. 2 shows the steps of RUSBagging SVM algorithm. At first, we use the Random Under-Sampling method to re-sample the original data sets D and get a rebalanced training data set D^*. Then we train an SVM classifier based on D^*. Repeat the process above for T times to make sure as many distinct samples as possible are sampled and then we get T SVM classifiers. After that, we integrate all these classifiers by calculate the weight over every row and column of each classifier.

A. Random Under-Sampling

To generate a balanced date set for training, we use the under-sampling method to re-sample the original set. Assuming that the size of the re-sampled data sets is S, where $S \leq N_p \times 2$, N_p is the size of the positive set P. We randomly sample $\frac{S}{2}$ number of instances from both of the positive set P and the negative set N without replacement and put them into the new training data set D^*.

To make sure that every subset $D^*{}_i$ for training are relatively independent and as many instances of the original sets as possible are covered, we present a concept of $overlap$ $rate$. The $overlap\ rate$ of two data sets is defined as follows:

Given two data sets D_1, D_2 with the size of N, N_r is the number of the same samples in D_1 and D_2, so the $overlap$ $rate$ R_0 of D_1 and D_2 is:

$$R_0\left(D_1, D_2\right) = \frac{N_r}{N} \tag{4}$$

We set the threshold value $R_{Threshold}$ to limit the subset D_t got from the tth under-sampling by:

$$R_0\left(D_t, D_i\right) < R_{Threshold}, i = 1, 2, ..., t-1 \tag{5}$$

B. Training SVM in Every Iteration

In every iteration to generate an SVM, we sampling the original data sets to make a subset for training under the limit of restricted condition above. After that, we need to train the parameter of linear SVM to make a good performance. In this paper, Algorithm 1 is used to choose the parameter of SVM.

Algorithm 1 Choosing the most optimal parameter of SVM

Require: Original training set $D\left\{(x_1, y_1), ..., (x_N, y_N)\right\}$. The subset $D_t\left\{(x_1, y_1), ..., (x_M, y_M)\right\}$in the t$th$ round of iteration. Where $x_i \in R^d, y_i \in \{-1, 1\}$,

Ensure: The most optimal parameter C_{opt} of SVM

1: Generate the validation set D_{val} by $D_{val} = D - D_t$.
2: For all C such that $C \in [0.01, 0.05, 0.1, 0.5, 1]$ do
3: Train the linear SVM classifier $h(x)$ on D_t by using the parameter C.
4: Predict the results of D_{val} by using $h(x)$. Calculate $Crit$ by:
$$Crit = \frac{TP}{TP + FP + FN}$$
5: end for
6: $C_{opt} = argmax_C Crit$

In Algorithm 1, the property of the classification is evaluated by:

$$Crit = \frac{TP}{TP + FP + FN} \tag{6}$$

Where TP is the number of positive samples that are correctly predicted, FP is the number of positive samples that are falsely predicted and FN is the number of negative samples that are falsely predicted. We don't use the accuracy rate to evaluate the performance, because we attach more attention to the classification effect of the smaller class, namely, the positive class. So the value of $Crit$ is more suitable to estimate imbalance data sets.

C. Integration of SVMs

The integration process of RUSBagging SVMs uses the Bagging algorithm. After T times of iteration, we weight every classifier by each row and column to do the integration, rather than simply add up all the decisions of classifiers. Assuming that the T classifiers we get from iteration are $h_1, h_2, ..., h_T$, firstly, we need to calculate the weights of every classifier by each row and column. Let $f_t(x_{r|c}) = Sign(h_t(x_{r|c}))$ represents the decision made by classifier h_t for the instance $x_{r|c}$ of whether the signal contains P300 potential or not when the $r|cth$ row or column is flashing. Where the instance x_i comes from the entire original training set. Then the weight of classifier h_t by each row and column can be calculated by:

$$w_{tr|c} = \frac{\sum_{i=1}^{N_{r|c}} I(f_t(x^i_{r|c}) == y^i_{r|c})}{N_{r|c}} \tag{7}$$

Where $t = 1, ..., T$, $r|c = 1, ..., 12$, $r|c$ represents the label of the six rows or six columns. $N_{r|c}$ is the number of the instances which represent the $r|cth$ row or column in the original training set. And $y^i_{r|c}$ is the correct label of sample x_i with the row or column number of $r|c$. Function $I(a == b)$ returns 1 when $a = b$ but returns 0 when $a \neq b$.

Then the linear weighting model of integration is:

$$H(x_{r|c}) = \frac{1}{T} \sum_{t=1}^{T} w_{tr|c} h_t(x_{r|c}) \tag{8}$$

Furthermore, in order to increase the accuracy of detection, 15 trials of experiments for one character is made. If we use $J(0 < J < 15)$ groups of data from these 15 trials, the final decision $S_{r|c}$ to predict a character is:

$$S_{r|c} = \frac{1}{J} \frac{1}{T} \sum_{j=1}^{J} \sum_{t=1}^{T} w_{tr|c} h_t(x^{(j)}_{r|c}) \tag{9}$$

$$= \frac{1}{J} \frac{1}{T} \sum_{j=1}^{J} \sum_{t=1}^{T} w_{tr|c} (\sum_{i \in D_t} y_i a^{(t)}_i \left\langle x^{(j)}_{r|c}, x_i \right\rangle + b^{(t)}) \tag{10}$$

$$= \frac{1}{T} \sum_{t=1}^{T} w_{tr|c} (\sum_{i \in D_t} y_i a^{(t)}_i \left\langle \frac{1}{J} \sum_{j=1}^{J} x^{(j)}_{r|c}, x_i \right\rangle + b^{(t)}) \tag{11}$$

Where $x^{(j)}_{r|c}$ is the instance at the jth sequence, x_i is the training set in the tth round of iteration, and $a^{(t)}_i, b^{(t)}$ are the parameters of the tth SVMs trained from data set D_t. The complete algorithm of RUSBagging SVMs is shown in Algorithm 2.

IV. Experiments and Results

A. The data sets we used in our experiments

In this paper, we use the data set IIb from the 2nd Wadsworth BCI competition in 2003[11] and the data set II from the 3rd Wadsworth BCI competition in 2005[12] as the data sets to exam the methods.

In the data set IIb from the 2nd Wadsworth BCI competition, the EEG data sets of only one subject are included. There are 3 groups of trials in the experiment, two of which are regarded as the training sets and the third group is the

Algorithm 2 RUSBagging SVMs

Require: Original training set $D\left\{(x_1, y_1), ..., (x_N, y_N)\right\}, x_i \in R^d, y_i \in \{-1, 1\}$. The maximum number of iterations T. The current round of iterations t;

Ensure: RUSBagging SVMs predicting model;

1: Initialize t to 1. Set the size of subset as M, where $M < N$, N is the size of original data set;

2: Randomly sample the original set D by using the method mentioned in Section 3.1. Then the subset D_t is generated, whose size is M;

3: Choosing the most optimal parameter C_{opt} of SVM based on D and D_t by using Algorithm 1;

4: Training the most optimal linear SVM classifier by using C_{opt} and D_t. The decision function in the tth round of iteration is:

$$h_t(x) = \sum_{i \in D_t} y_i a_i^{(t)} \langle x, x_i \rangle + b^{(t)}$$

5: Calculate the weight $w_{tr|c}$ of every classifier by each row and column, where $r|c = 1, ..., 12$;

6: If $t \leq T$, t=t+1, turn to step 2. Else turn to step 7;

7: Integrate all the SVM classifiers to make a final decision by:

$$H(x_{r|c}) = \frac{1}{T} \sum_{t=1}^{T} w_{tr|c} h_t(x_{r|c})$$

predicting set. Finally, in our examining program, the training set contains EEG data from spelling trials of 39 characters, and the predicting set contains EEG data from 31 characters.

The data set II from the 3rd Wadsworth BCI competition consists of the EEG data sets of two subjects A and B. Each subject is asked to execute the spelling task of 85 characters for the training set and 100 characters for the predicting set. Finally, we get four data sets, including two training sets and two predicting sets.

B. The results of experiments and comparison

In the experiments, we set the maximum number of iterations T as 100 and size of subset in every iteration is 900.

Fig. 3 (a) shows the results on data set IIb from the $2nd$ Wadsworth BCI competition by using RUSBagging SVMs. Fig. 3 (b) and Fig. 3 (c) shows the results on the data set of subject A and B in the data set II from the 3rd Wadsworth BCI competition. The horizontal axis represents the number of iterations and the vertical axis represents the accuracy rate. These figures present the performance of RUSBagging SVMs on data sets when using data from different number of epochs. In experiment(a), the data of 1 to 4 sequences of repeated trials is used, while in (b) and (c), the data of 1,5,10,15 sequences is used.

Then we make a comparison with a single linear SVM[13]. The best parameter of SVM is trained from cross validation. Because the subset in every time to train an SVM in RUSBagging SVMs algorithm is randomly sampled from the original data sets, to get a more convincing result, the average results of several experiments are used to make a comparison. The

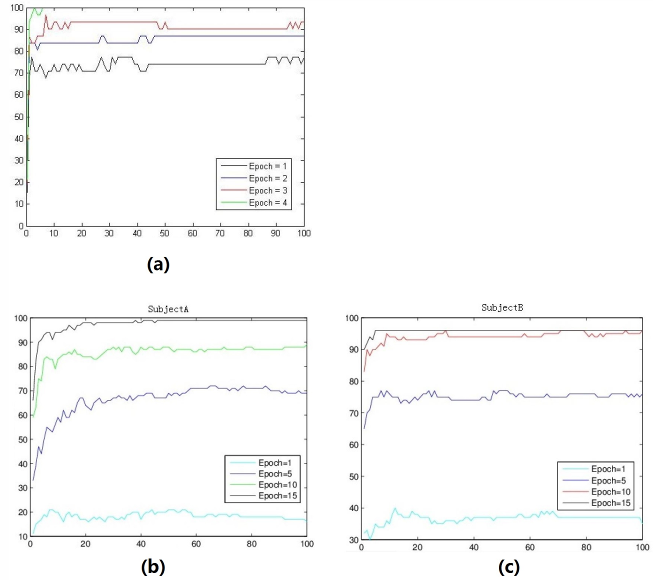

Fig. 3. Figure(a) shows the performance of RUSBagging SVMs on data set IIb from the $2nd$ Wadsworth BCI competition. Figure(b) and (c) show the performance of RUSBagging SVMs on the data set of subject A in the data set II from the $3rd$ Wadsworth BCI competition.

TABLE I. THE ACCURACY RATE/ITR(%/(BITS/MIN)) BY USING A SINGLE LINEAR SVM AND RUSBAGGING SVMS ALGORITHM BASED ON DATA SET IIB FROM THE 2ND WADSWORTH BCI COMPETITION.

Method	Number of Sequences						
	1	2	3	4	5	10	15
RSVMs	80.3/44.9	87.1/35.4	92.3/29.9	100.0/28.5	100.0/23.9	100.0/13.2	100.0/9.1
SVM	77.4/42.3	87.1/35.4	90.3/28.7	96.8/26.4	100.0/23.9	100.0/13.2	100.0/9.1

accuracy and ITR of these two methods based on data set IIb from the 2nd Wadsworth BCI competition are shown in Table 1. Here ITR means the amount of information transmitted in unit time by a BCI system, and is calculated by[14]:

$$ITR = \frac{60(Plog_x(P) + (1-P)log_2(\frac{1-P}{N-1}) + log_2(N))}{T}$$

From Table 1, we can find that the accuracy rate of RUSBagging SVMs reaches to 100% when using the data of 4 trials of stimulation experiments but linear SVM needs 5. Besides, the ITR of RUSBagging SVMs is generally higher than linear SVMs.

Then the data set II from the 3rd Wadsworth BCI competition is used to make a comparison with these two methods and the results of accuracy and ITR are shown in Table 2. From the results, we can see that RUSBagging SVMs algorithm has a better performance than linear SVM in most conditions when using different number of sequences. What's more, the experiments are conducted in computer with the memory of 6GB. If we set the number of iterations T as 100, RUSBagging SVMs algorithm costs nearly 40 minutes for training. However, linear SVM costs almost 6 hours. While the time cost for predicting of these two methods is almost the same.

Furthermore, we make comparisons with other competitors in the $2nd$ and $3rd$ Wadsworth BCI competitions, whose methods ranked in the top.

The comparison results with the methods based on the $2nd$

TABLE II. THE ACCURACY RATE/ITR(%/(BITS/MIN)) BY USING A SINGLE LINEAR SVM AND RUSBAGGING SVMS ALGORITHM BASED ON DATA SET II FROM THE 3RD WADSWORTH BCI COMPETITION.

Subject	Method	Number of Sequences					
		1	2	3	5	10	15
A	RSVMs	16.2/3.0	34.5/7.9	56.7/13.4	68.1/12.1	83.7/9.4	97.3/8.6
	SVM	16.0/3.0	34.0/7.7	49.0/10.6	58.0/9.4	83.0/9.3	98.0/8.7
B	RSVMs	36.8/12.7	57.8/18.1	68.2/18.0	78.7/15.4	95.7/12.0	96.1/8.4
	SVM	45.0/17.7	63.0/20.8	69.0/18.3	81.0/16.1	93.0/11.4	95.0/8.2
Mean	RSVMs	26.5/7.4	46.1/12.6	62.4/15.6	73.4/13.7	89.7/10.6	96.7/8.5
	SVM	30.5/9.4	48.5/13.7	59.0/14.3	69.5/12.6	88.0/10.3	96.5/8.4

TABLE III. THE NUMBER OF WRONGLY RECOGNIZED CHARACTERS BASED ON THE DATA SET IIB FROM THE $2nd$ WADSWORTH BCI COMPETITION BY USING DIFFERENT METHODS.

Method	Number of Sequences					
	1	2	3	4	5	6
Kaper[15]	12	8	8	2	1	0
Bostanov[16]	11	5	2	1	1	0
Blankertz[17]	16	14	6	3	2	0
RUSBagging SVMs	5-7	4	2-3	0	0	0

TABLE IV. THE ACCURACY RATE/ITR(%/(BITS/MIN)) BY USING DIFFERENT METHODS BASED ON DATA SET II FROM THE 3RD WADSWORTH BCI COMPETITION. NOS MEANS NUMBER OF SEQUENCES.

Subject	NoS	Method				
		LDA[18]	ESVM[19]	CNN-1[14]	MCNN-1[14]	RSVMs
A	5	45/6.26	72/13.28	61/10.18	61/10.18	68.1/12.14
	10	78/8.38	83/9.29	86/9.87	82/9.11	83.65/9.42
	15	88/7.1	97/8.51	97/8.51	97/8.51	97.3/8.56
B	5	76/14.51	75/14.2	79/15.47	77/14.83	78.70/15.37
	10	92/11.13	91/10.91	91/10.91	92/11.13	95.6/11.97
	15	96/8.33	96/8.33	92/7.69	94/8	96.1/8.35
Mean	5	60.5/10.04	73.5/13.74	70/12.69	69/12.4	73.4/13.71
	10	85/9.68	87/10.07	88.5/10.38	87/10.07	89.65/10.62
	15	92/7.69	96.5/8.42	94.5/8.08	95.5/8.25	96.7/8.46

BCI competition data sets are shown in Table 3. The table records the number of characters that are wrongly recognized when using different number of sequences. From the table, we can see that by using RUSBagging SVMs, all the characters are correctly recognized at the $4th$ sequence, while other methods need more sequences.

Then we compared with the methods that performed well in the 3rd BCI competition. The data sets on this competition are used and the results are shown in Table 4. The table shows the accuracy rate (%) and ITR(bits/min) of these methods when using data of 5,10,15 sequences of repeated trials. In [14], the methods by using Convolutional Neural Networks for P300 detection are shown to be effective from the results, so we compared with them as well. Among the methods used for comparison, ESVM[19] was considered as one of the most effective methods and ranked the top of the $3rd$ BCI competition. From the table, we can see that RUSBagging SVMs are very competitive in terms of both accuracy rate and ITR compared to other methods, especially when using more sequences of trials.

V. CONCLUSION

The P300 EEG data sets are naturally imbalanced because of the conditions to evoke P300 potential and the property of P300 Speller Diagram. In this paper, the algorithm of RUSBagging SVMs is proposed to detect the P300 potential taking account of the imbalance problem in P300 EEG data sets. Then several character recognition experiments based on P300 EEG are conducted to test the performance of RUSBagging SVMs. From the experiments, RUSBagging SVMs are proved to be an effective method to detect P300 potential. It can also be concluded that the imbalance problem indeed has influence on the P300 detection and improvements can be made if the problem are fully considered and properly solved.

ACKNOWLEDGMENT

This work is supported in part by the National Science Foundation of China under Grant Nos. (61375064, 61373001) and Jiangsu NSF grant (BK20131279).

REFERENCES

[1] Haselsteiner, E., and Pfurtscheller, G. ”Using time-dependent neural networks for EEG classification.” Rhablaon ngnrng ranaon on 8.4(2000):457 - 463.

[2] N. Masic and G. Pfurtscheller, Neural Network Based Classification of Single-Trial EEG Data, Artificial Intelligence in Medicine, vol. 5, pp. 503513, 1993.

[3] A. Rakotomamonjy and V. Guigue, BCI Competition III: Data Set II Ensemble of SVMs for BCI p300 Speller, IEEE Trans. Biomedical Eng., vol. 55, pp. 11471154, Mar. 2008.

[4] Xu N, Gao X, Hong B, et al. BCI competition 2003-data set IIb: enhancing P 300 wave detection using ICA-based subspace projections for BCI applications[J]. IEEE transactions on biomedical engineering, 2004, 51(6): 1067-1072.

[5] Kim, Min Ki, and Sung-Phil Kim. ”Detection of P300 components using the Wiener filter for BCI-based spellers.” Control Conference (ASCC), 2011 8th Asian. IEEE, 2011:892 - 896.

[6] Bostanov, V. ”BCI competition 2003-data sets Ib and IIb: feature extraction from event-related brain potentials with the continuous wavelet transform and the t-value scalogram.” ranaon on Bomdal ngnrng 51(2004):1057 - 1061.

[7] Li, Peng, Qiao Pei-li, and Yuan-Chao Liu. A hybrid re-sampling method for SVM learning from imbalanced data sets. Fifth International Conference on Fuzzy Systems and Knowledge Discovery, Vol. 2. IEEE, 2008.

[8] Duncan-Johnson C C, Donchin E. On Quantifying Surprise: The Variation of Event-Related Potentials With Subjective Probability[J]. Psychophysiology, 1977, 14(5):456467.

[9] Gu, Q., Cai, Z., Zhu, L., & Huang, B. (2008). Data Mining on Imbalanced Data Sets. Advanced Computer Theory and Engineering, 2008. ICACTE ’08. International Conference on (pp.1020 - 1024). IEEE.

[10] L. Farwell and E.Donchin, Talking off the top of your head: Toward a mental prosthesis utilizing event-related brain potentials, Electroencephal. Clin. Neurophysiol., vol. 51, pp. 1034-1043, 2004.

[11] BCI Competition II Webpage. http //www.bbci.de/competition/ii/, 2003.

[12] BCI Competition III Webpage. http //www.bbci.de/competition/iii/, 2005.

[13] Cortes, C., Vapnik, V. Support vector machine. Machine learning, 20(3): 273-297, 1995.

[14] Cecotti H, Graser A. Convolutional neural networks for P300 detection with application to brain-computer interfaces[J]. Pattern Analysis and Machine Intelligence, IEEE Transactions on, 2011, 33(3): 433-445.

[15] Matthias Kaper, Peter Meinicke, Ulf Grossekathoefer, Thomas Lingner and Helge Ritter. BCI Competition 2003?Data Set IIb: Support Vector Machines for the P300 Speller Paradigm. IEEE TRANSACTIONS ON BIOMEDICAL ENGINEERING, Vol. 51, No. 6, June 2004.

[16] Bostanov V. BCI competition 2003-data sets Ib and IIb: feature extraction from event-related brain potentials with the continuous wavelet transform and the t-value scalogram[J]. ranaon on Bomdal ngnrng, 2004, 51:1057 - 1061.

[17] Blankertz, Benjamin, et al. The BCI competition 2003: progress and perspectives in detection and discrimination of EEG single trials. IEEE Transactions on Biomedical Engineering, 51(6): 1044-1051, 2004.

[18] N. Liang and L. Bougrain. Averaging Techniques for Single-Trial Analysis of Oddball Event-Related Potentials. Proc. Fourth Int’l BCI Workshop and Training Course, pp. 44-49, 2008.

[19] B. Blankertz. BCI Competition III Final Results. 2008.