Prediction of the Admission Lines of College Entrance Examination based on machine learning

Zhenru Wang
State Key Laboratory of Networking and Switching Technology
Bei Jing, 100876, China
e-mail: wangzr622@gmail.com

Yijie Shi
State Key Laboratory of Networking and Switching Technology
Bei Jing, 100876, China
e-mail: yijieshi2000@bupt.edu.cn

***Abstract*—Accurate prediction to college entrance examination(CEE) results is very important for the candidates to fill in the application and the relevant analysis of the CEE. At present, the prediction of CEE scores is based on data statistics, probability model and some weighted combination models. Since generating the model for predicting college admission lines uses too little reference factor, and the error is relatively large, so the reference value is very small. In this paper, machine learning methods are used to carry out the college admission lines of research and prediction. Specially, in this paper Adaboost algorithm is used to study and forecast, which belongs to ensemble learning . Finally, the result of this model is given, which is better than the current prediction method.**

Keywords-adaboost; machine learning; college entrance examination; admission line forecast

I. BACKGROUND

Every year, the college entrance examination, as an important way for the country's selection of elite, is very important for each of the Chinese mainland students and their parents, which may affect the fate of themselves.

In the face of the annual college application, a lot of people refer to an admission line, which was the old standard of last year. So the accuracy of prediction greatly affects whether the students could reach the college that they want to. Prediction to admission line is very important.

At present, forecasting admission lines in many schools mainly based on the past rates of admission over the years, and the students' estimate scores. Of course, the premise of this method is assuming that the academic standards of the schools remained stable, and the admission rate would not have a big change. However, if there is some irresistible factors, such as some changes of enrollment policy, or the difficulty of the examination changing, then the admission lines may differ by more than a dozen or even dozens of points.

Besides this method the variable weight combination model, which combine Power Model, Logrithm model and Linear Model, was proposed to forecast the admission lines.Each model has a different weight, and these models are combined into a new model . In addition, the Norm Model was proposed to predict the results, and this method depended on the government documents. However, these models are simple mathematical model, only take few factors into account, and cannot fully reflect the influence of many factors, such as the number of candidates, the degree of difficulty of the examination, and so on. Therefore, in this paper we present a method of using machine learning.

Tom Mitchell, in the book Machine Learning, thinks that the concerned problems of machine learning are how the systems automatically improve the performance with the increase of experience, (e.g., programs that learn to recognize human faces, recommend music and movies, and drive autonomous robots). In this paper , one machine learning algorithm named Adaboost[2][7][8] is applied. The traditional machine learning method is in a space which is composed of a variety of possible functions to find a closest to the actual classification function f, which is classifier h[1][3]. Now, the single classifier models mainly include decision tree, artificial neural network, Naive Bayes classifier[1] and so on. Ensemble learning integrates a number of individual classifiers. The final result was determined by an ensemble classifier, which is combined by several single classifiers, and proved to have a better effect.

In this paper, we use machine learning methods to study and predict the admission lines of the CEE, and finally get a good prediction model. This paper is organized as follows: the section two is mainly about the algorithm Adaboost used in this paper. In the section three, the selection of the features is described, which is used for Adaboost algorithm. In the fourth part, a series of experiments is introduced, which is used to demonstrate the accuracy of the prediction results. And finally,a conclusion is given about deficiencies of the experiment.

II. ALGORITHM INTRODUCTION

Adaboost algorithm is an iterative algorithm, which is a strong classifier composed of different weak classifiers. The weights of each sample are determined by whether the samples are classified correctly, and the correct rate of the overall classification in the last time. Then, the new sample data with weights is transferred to the lower level classifier to train. At this time, the classifiers we got are weak classifiers. Finally, the weak classifiers are obtained by several iterations, and a final classifier is obtained by the voting strategy.

The whole process is as follows:

1) Firstly, study N samples, and obtain the first weak classifier.
2) A new group of N training samples is composed of the wrong samples and the other new data. The second weak classifiers are obtained through the learning of this sample group.
3) The wrong samples are divided in step 1 and step 2 with other new samples, which constitute a new group of N training samples. Then, the third weak classifier was obtained by learning these samples.
4) Finally, a strong classifier is generated, which is combined by several weak classifiers.

A detailed algorithm is shown in figure 1.

Input :
1 D, a set of *d* class-labeled training tuples;
2 *k*, the number of rounds (one classifer is generated per round);
3 a classification learning scheme
Output: a composite model.
Method:
(1) Initialize the weight of each tuple in D to $1/d$;
(2) **for** i = 1 to k **do:**
(3) sample D with replacement according to the tuple weights to obtain D_i;
(4) use training set D_i to derive a model, M_i;
(5) compute $error(M_i)$, the error rate of M_i
(6) **if** $error(M_i) > 0.5$ **then**
(7) reinitialize the weights to $1/d$
(8) go back to step 3 and try again;
(9) **endif**
(10) **for** each tuple in D_i that was correctly classfied **do**
(11) multiply the weight of the tuple by $error(M_i)/(1-error(M_i))$;
(12) normalize the weight of each tuple;
(13) **endfor**

Figure1. The Adaboost algorithm

In the training phase, each training samples group weights initially equal to 1/d, where d is the number of samples. Then, every time we use a part of training samples to train the weak classifier, and only keep the weak classifiers, of which the wrong rate is less than 0.5. For the training samples that had been divided into correct class, the weights are adjusted to $error(M_i)/(1-error(M_i))$, where $error(M_i)$ is the error rate of $i-th$ weak classifier (reducing the weights of correctly classified samples is to increase the weights of fault samples). The selection of the classifier is shown in figure 2.

To use the composite model to classfy tuple, *X*:
(1) Initialize weight of each class to 0;
(2) **for** i = 1 to k **do**
(3) $w_i = log\frac{1-error(M_i)}{error(M_i)}$;
(4) $c = M_i(X)$;
(5) add w_i to weight for class c
(6) **endfor**
(7) return the class with the largest weight;

Figure 2. The selection of the Adaboost classifiers.

In the stage of selecting classifier, each weak classifier gives its own prediction results, and the weight of the weak classifier is:

$$\log(error(M_i)/(1-error(M_i))).$$

The result of the highest weight of a weak classifier becomes the final prediction result.

Random forests [6][9][10] algorithm is also a kind of ensemble learning algorithm. It also has a number of weak classifiers and these weak classifiers compose a strong classifier. And it can also be used for multi-classification and regression, but random forests algorithm is composed by multiple decision trees, unlike AdaBoost. And in this paper, we use random forests algorithm as a comparison of AdaBoost.

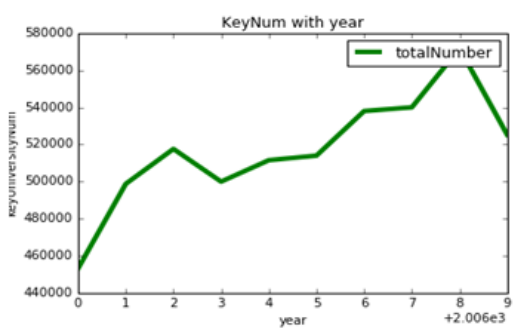

Figure 3. The number of applicants with the time change in province Sichuan

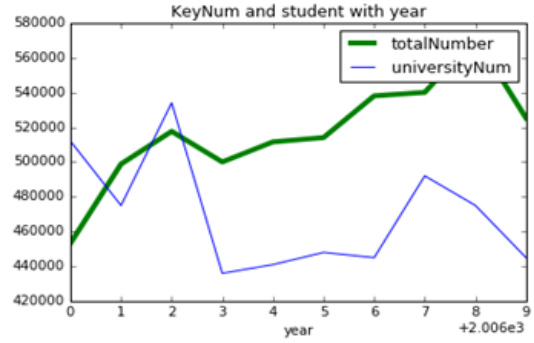

Figure 4. Comparison between indicates of CEE and the admission line of second batch of undergraduate.

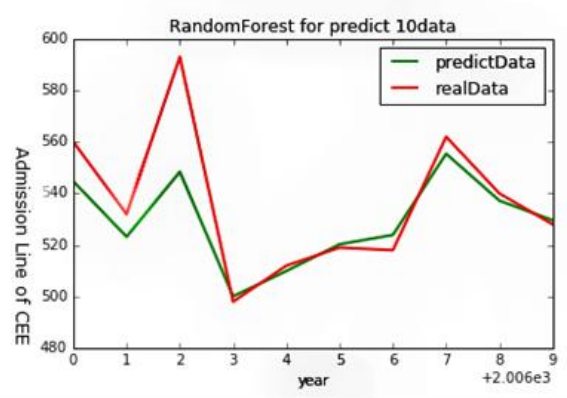

Figure 5. Using random forests to learn the ten year's data and forecast.

III. FEATURE SELECTION

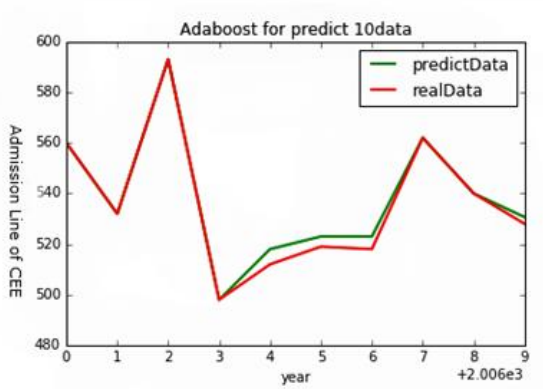

Figure 6. Using Adaboost to learn ten year's data and forecast

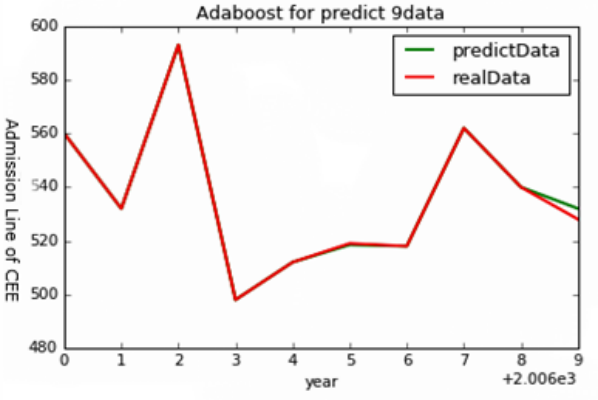

Figure7. Using Adaboost to learn nine year's data and forecast

For machine learning, the feature selection is a very important job. The quality of features is directly related to the quality of the model, and will affect the accuracy of the predicted results. In this paper, we select the following features, and achieve a good result.The selected features are listed as following. The data were selected from 2006 to 2015 in Sichuan Province.

TotalNumber: total number of indicates of CEE in province Sichuan. In figure 3, the total number of students attending the examination from year 2006 to year 2015 in province Sichuan is listed. In this figure, we can find that the number of applicants is increasing from 2006 to 2014, unless 2007 and 2009. And the trend of increment is less and less.

NumberOfKeyUniversity: enrollment plan of first batch of undergraduate. Because the admission line of first batch of undergraduate is based on the enrollment plan of first batch of undergraduate. So this feature is important to modeling and forecasting.

NumberOfUniversity: enrollment plan of second batch of undergraduate. Same as last feature, every year the admission line of second batch of undergraduate is based on the enrollment plan of second batch of undergraduate.

NumberOfMath: the number of science students who take part in CEE.

Param: difficulty of test question. In this paper, the ratio of the total number of enrollment plan of second batch of undergraduate and the total number of enrollment plan is selected as the estimated value of the difficulty of the test questions. In figure 4, the comparison between indicates of CEE and the admission line of second batch of undergraduate. Obviously the trend of the admission line of second batch of undergraduate is same as the trend of indicates of CEE, except some special years. So, it is meaningful to select this feature.

IV. EXPERIMENT DESCRIPTION

Firstly, we select Adaboost to conduct an experiment, and then, we conduct an experiment with Random Forests. By contrast, it is obvious to know which ensemble learning algorithm has a better result with the real data, and has better application conditions for the CEE.

In the experiment, the data were selected from 2006 to 2015 in Sichuan Province. The experimental environment is as follows: the system is Windows 7, and Python is used to program.

In this paper, two groups of comparative experiments were conducted. One experiment uses Adaboost to learn different numbers of samples, then constructs the model to predict all the data, including the learned and non-learned data. The other experiment uses Random Forests algorithm and Adaboost algorithm to learn the same data respectively, and then use the model to predict. The experimental results are as follows:

In figure 5 and figure 6, by using 10 years data to train and making a prediction, the results of Random Forests and Adaboost method are respectively showed. The prediction accuracy of Random Forests was over 80% while the accuracy of Adaboost was over 90%, and using Adaboost is much better than Random Forests. It can be seen from the figures that the Random Forests prediction has a certain deviation from the actual value, but the trend, increase or decrease for every year, is correct. Using Adaboost, only having a slight difference in the individual years, has tiny error which is within 5 points.

From this experiment, it can be seen that Adaboost had a good effect than Random Forests. The reason is that Adaboost is a strong classifier constructed by several weak classifiers. So it has a good effect on complex data, such as the data of college entrance examination. While the Random Forests algorithm processes the complex data by subdividing the space. Therefore, Adaboost is a better model to handle college entrance examination problem.

Figure 7 showes using Adaboost to learn nine years' data and predict the year that had not to learn. It can be concluded that Adaboost, only having 4 point gap with the real data, has a good effect.

Comparing figure 7 and figure 6, it can be found that even if the data learned by Adaboost algorithm is reduced, the accuracy of the model will be still very high.

V. CONCLUSION

Through the above experiment and analysis, it can be concluded that AdaBoost prediction model has a good effect on predicting the admission line of CEE. The correctness of result is verified by experiment. This model considers the enrollment plan, number of applicants, the difficulty of test

question, and other factors. The result of this prediction model has important reference value to the students who take part in college entrance examination.

Of course, the establishment of the model is not very perfect because of the in-exhaustive data. And there are still a lot of things to be improved. And in the aspect of feature selection, we only forecast the college entrance examination of province Sichuan. If getting more data in the future, we also can do universities admission line forecast. It is also a very significance work.

ACKNOWLEDGMENT

This work is supported by NSFC (Grant Nos. 61300181, 61502044), the Fundamental Research Funds for the Central Universities (Grant No. 2015RC23).

REFERENCES

[1] Tom. Mitchell. Machine Learning[M]. 2003.

[2] Freund. Y & R. E. Schapire. Experiments with a new boosting algorithm[J]. Machine Learning: Thirteenth International Conference, 1996

[3] homas, G, Dietterich. Ensemble learning[J]. Handbook of Brain Theory and Neural Networks, 2002, (2)

[4] Thomas. G. Dietterich. Ensemble Methods in Machine Learning[J]. Multiple Classifier Systems, 2000, 1857: 1-15

[5] G. Valentini & F. Masulli. Ensembles of Learning Machines[J]. Workshop on Neural Nets, 2002, (2486): 3-20

[6] L. Breiman. Random Forests[J]. Machine Learning, 2001, 45(1): 5-32

[7] Y. Freund, R. Schapire, A Decision-Theoretic Generalization of on-Line Learning and an Application to Boosting[J]. Journal of Computer and System Sciences, 1997, 55: 119-139

[8] H.Drucker, Improving Regressors using Boosting Techniques[J]. ICML,1997,107-115

[9] P. Geurts, D. Ernst., and L. Wehenkel, Extremely randomized trees, Machine Learning[J], 63(1), 3-42, 2006.

[10] Moosmann, F. and Triggs, B. and Jurie, F. Fast discriminative visual codebooks using randomized clustering forests[J], NIPS ,2007