Membrane Technology
ISSN (online): 1873-4049

Research on Prediction of College Students' Registration Based on Machine Learning and Voting Model

Lei Yang[1,2,*], Shipei Du[1], Yungui Chen[1], Longqing Zhang[1,2]

[1]*Guangdong University of Science and Technology, Dongguan, China*

[2]*Dongguan Key Laboratory of Industrial Software for Intelligent Manufacturing, Dongguan, China*

**Corresponding Author.*

Abstract

The registration rate of freshmen has always been a major concern for many colleges and universities, especially private ones. The author of this paper collects and completes a dataset of freshmen registration and uses various machine learning algorithms, including decision tree, random forest, and BP neural network, to learn from it. The author introduces the confusion matrix and F1 score to evaluate the effect of machine learning. A voting model based on multiple machine learning algorithms is designed to optimize prediction and the effectiveness of this scheme is verified through numerous experiments. The experimental results also reveal the factors affecting the registration of college freshmen, which has certain guiding significance for the enrollment of colleges and universities. 75% of the data is used as the training set and 25% as the test set. The data is preprocessed to make it standardized and complete. The results show that the performance of the voting model is significantly improved compared to a single algorithm, and the prediction accuracy of freshmen registration is maintained at more than 60%, with the F1 score reaching 0.7.

Keywords: prediction, freshmen, voting model, machine learning.

INTRODUCTION

In universities, predicting the enrollment of freshmen is very difficult but essential work. In September every year, the management department of the university will arrange all work in advance before the arrival of freshmen. For example, the planning of student dormitories, the construction of student activity venues, the recruitment of new teachers, the purchase of new books in the library, etc., all of which are related to the real number of freshmen registered. If these deployments are not well prepared before the start of school, it is likely to cause chaos in the daily order of the University and even affect the reputation of the University.

In order to solve this problem, this research collected one Chinese University's entrance examination admission data and freshman enrollment data in recent years. Based on these data, the author created a freshman registration dataset for machine learning. This research team is one of the few research teams that use machine learning to study whether freshmen will register. Based on the team's previous research, it has been confirmed that this thing is predictable. In this paper, the main contribution is to provide an optimization scheme to improve the prediction performance. The optimization scheme is that this research establish a voting model and use this voting model to predict. The voting model is composed of three machine learning algorithms: decision tree, random forest, and BP neural network. Hard voting and soft voting are used to predict respectively. This research have verified the effectiveness of this scheme through a large number of experiments, and the performance is improved obviously.

In the classification problem [1-3], the decision tree can be regarded as a combination of if-then rules, or as a conditional probability distribution in a specific feature space and class space. Decision tree learning is mainly divided into three steps: feature selection, decision tree generation, and pruning operation. In the research of many classification problems, the decision tree is often one of the preferred machine learning algorithms.

The general method of random decision forests [4-6] was first proposed by Ho in 1995. Ho established that forests of trees splitting with oblique hyperplanes can gain accuracy as they grow without suffering from overtraining, as long as the forests are randomly restricted to be sensitive to only selected feature dimensions. A subsequent work along the same lines concluded that other splitting methods behave similarly, as long as they are randomly forced to be insensitive to some feature dimensions. Note that this observation of a more complex classifier (a larger forest) getting more accurate nearly monotonically is in sharp contrast to the common belief that the complexity of a classifier can only grow to a certain level of accuracy before being hurt by overfitting. The explanation of the forest method’s resistance to overtraining can be found in Kleinberg’s theory of stochastic discrimination.

Literature [7] proposes a cost sensitive random forest algorithm, which introduces cost sensitive learning into the random forest algorithm to make the classifier more biased to a few classes and minimize the total misclassification cost. Literature [8] first proposed NCL (neighborhood cleaning rule) processing for the original data and combined the processed data with a random forest algorithm for classification. The experiment shows that the random forest algorithm improved by NCL technology has better classification accuracy. Reference [9] proposed a stratified sampling random forest algorithm and used the support vector machine algorithm as the base classifier at the node splitting. The results show that the improved random forest algorithm has a better processing effect on unbalanced data than the traditional random forest, oversampled support vector machine, and undersampled support vector machine. This research improve the algorithm by adjusting the number of trees in the random forest algorithm, and the results show that it is effective.

BP (backpropagation) neural network is a multilayer feedforward neural network trained according to the error backpropagation algorithm [2, 10, 11]. In terms of structure, the BP network has an input layer, hidden layer, and output layer. The BP algorithm uses the square of the network error as its objective function and uses the gradient descent method to calculate the minimum value of the objective function. These neurons are called to hide cells; they do not pertain to the external world immediately, but changes in their status can influence the association between the output and input. Incidentally, each layer has multiple neurons.

Finally, according to this research, they reveal the factors affecting the registration of freshmen, which has a certain guiding significance for the enrollment of colleges and universities.

THE DATASET PREPROCESSING

The data used in this paper comes from a university in Guangzhou. This research collected the data of this university in recent 6 years. These data are from the official information of the college entrance examination admission system and the internal data of the University's own information management system. These data are authorized by the University for research purposes only. Because some data involves students' personal privacy, this research have fuzzed this part of the data and deleted some unnecessary data reader don't care about.

The whole dataset contains the data of this university in the past six years. The total data volume is 17652, of which 9839 students are enrolled and 7813 students are not enrolled. This research use 75% of the data as the training set and the remaining 25% of the data as the test set, that is, the number of samples in the training set is 13239 and the number of samples in the test set is 4413. It can be seen that this dataset is a relatively balanced dataset, which is very important for the binary classification problem the authors study. For those schools with high enrollment rates, their dataset may not be a balanced dataset, which needs to be optimized from the selection of data sampling methods and algorithms.

Before machine learning these collected data, the first thing the authors need to do is preprocess these data [12-14]. Because most of the collected raw data are missing, some data are noisy, and even some data will be repeatedly collected, these collected raw data can not be used directly. A large amount of processing work needs to be carried out on these raw data to make them become data that can be recognized and used by the program.

In the new dataset the authors constructed, the values of some feature attributes are null. For example, in the art score column, some students do not have this score, so they need to change the null value to 0. There are many similar feature attributes. The authors have done the same treatment, that is, change the null value to 0.

Then there are some inconsistent data [15]. For example, the name of a major in a university often changes. For example, there is a major called computer network technology, which was changed to computer network a few years ago. These majors are essentially the same major, the authors have standardized their names in the past few years.

In addition, some noise data need to be removed. In this new dataset, some data are not related to this research content, such as the names of students' parents, headteachers, etc. another additional piece of data that needs to be paid attention to is the number of students from other provinces, which is very small, less than 0.5% of the cases, and these students from other provinces are not within the scope of this research, It is also necessary to remove these data.

Data reduction [16] is to minimize the amount of data on the premise of maintaining the original appearance of the data as much as possible. Because the source data in the new data table is very complete and contains a lot of repeated information, in order to reduce the time of program calculation and improve the efficiency of data learning, after analyzing the admission information and registration information, the authors deleted part of the data to reduce the amount of data in machine learning. For example, you only need to keep one of the registered permanent residences, mailing addresses, date of birth, and age.

During machine learning of datasets, many data types cannot directly participate in the calculation of the program, such as text data such as major name, student class, and student native place. Therefore, the authors need to convert the data types of these data to enable the program to calculate [1, 17, 18].

For example, one of the feature attributes is called fixed telephone, which was originally a digital string. The authors changed it to numbers 0 and 1. The number 0 indicates that the fixed telephone is not installed in the student's home, and the number 1 indicates that the fixed telephone is installed in the home. For another feature attribute mobile telephone, the authors did the same data conversion. In addition, they also convert some other text numbers into numerical numbers.

After the above processing methods, the dataset becomes standardized and complete, and the volume is reduced from 38 columns to 18 columns. The dataset after processing is as shown in Table 1.

Table 1. Dataset

	y	a_1	a_2	...	a_{14}	a_{15}	a_{16}
	Registration status	Gender	Score	...	School type	Examination type	Politic countenance
x_1	0	2	305	...	2	1	13
x_2	1	2	372	...	2	2	1
x_3	0	1	358	...	1	1	1
...	...	...	...	...	...	...	...

The symbol y denotes the registration status, i.e., 1 is registered, and 0 is not registered. The other symbols are defined as follows:

Attribute set A = $\{a_1, a_2, a_3, \ldots a_{15}, a_{16}\}$

Sample $x_i = \{a_1^i, a_2^i, a_3^i, \ldots a_{15}^i, a_{16}^i\}$

Samples set X = $\{x_1, x_2, x_3, \ldots x_{17651}, x_{17652}\}$

Training set $D = \{(x_1, y_1), (x_2, y_2), \ldots, (x_{13239}, y_{13239})\}$

e.g. (x_1, y_1) = (2 305 ... 2 1 13, 0)

METHODS

The Machine Learning Algorithms

A decision tree is a predictive analysis model of a tree structure that reflects the mapping between objects and their attribute values. It comprises a root node, branch node, and leaf node. The latter is the starting point of the entire decision tree and is located at the top. The branch node is a new attribute formed by dividing an upper node, representing a data subset of data. The leaf node represents the classification result. The decision tree judges from the root node and selects the node according to the attribute value of the upper node in a top-down manner until the leaf node forms a new class. Each path of the decision tree from the root node to the leaf node is a predictive path that visually represents the relationship between attributes and results.

With the growth of the decision tree, in order to classify all samples as accurately as possible, the operation of branching the tree will be repeated, and finally, a very huge decision tree model will be generated. The depth of this tree may be as high as hundreds of layers, but such a decision tree is not a good classification model.

The main reason is that the classification of training data in this model is too "accurate". With the increase of the number of layers of the tree, the number of samples of nodes is decreasing. The deeper the node, the more accurate its characteristics are. Although this overlearning accurately reflects the characteristics of the training data, it has no generalization ability. This is what the researchers often call overfitting, so the authors must deal with this huge tree to improve the generalization ability of the model. In order to solve the problem of excessive fitting of decision tree model, the authors propose two solutions in reference [19], which will not be repeated here.

A random forest is a classifier that uses multiple decision trees to train and predict samples. In particular, trees that are grown very deep tend to learn highly irregular patterns: they overfit their training sets, i.e. have low bias, but very high variance. Random forests are a way of averaging multiple deep decision trees, trained on different parts of the same training set, with the goal of reducing the variance. This comes at the expense of a small increase in the bias and some loss of interpretability, but generally greatly boosts the performance in the final model.

Each tree in the random forest algorithm grows to the greatest extent, and there is no pruning process [20]. The training set of each tree is randomly selected. If there is no random sampling, the training set of each tree is the same, then the classification result of the finally trained tree is exactly the same. In addition, the feature extraction of the sample is also randomly selected, Assuming that the feature dimension of each sample is N, specify a constant n <N, randomly select n feature subsets from N features and select the best from these n features each time the tree is split. The "randomness" in the random forest refers to this two randomness. The introduction of these two randomness is very important to the classification performance of random forests. Because of their introduction, random forest is not easy to fall into overfitting and has good anti-noise ability. Therefore, the results of machine learning are also random. In order to stabilize the randomness as much as possible, the authors calculated 10 times and finally took the mean as its final prediction result.

In order to further improve the accuracy of the prediction, the authors continuously increase the number of trees and use 10, 50, 100, 200, 500 trees to construct random forests respectively. The results show that more trees, more better.

BP (back propagation) neural network is a multi-layer feedforward neural network trained according to the error back propagation algorithm. It adds several layers (one or more layers) of neurons between the input layer and the output layer [21]. These neurons are called to hide cells, they are not directly related to the outside world, but their state changes can affect the relationship between input and output. Each layer can have several nodes.

In this paper, the authors created a neural network with three hidden layers, each containing 10 neurons. The constructed BP neural network is shown in Figure 1.

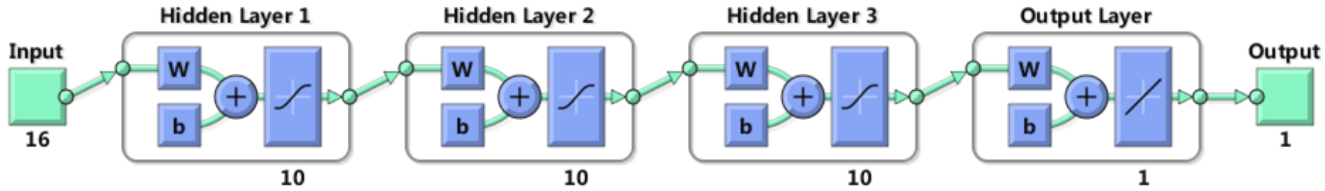

Figure 1. The BP neural network

Performance Metric

For the binary classification problem, the sample can be categorized into four cases: true positive (TP), false positive (FP), true negative (TN), and false negative (FN), according to the combination of its real category and the classifier prediction category [22, 23]. The TP, FP, TN, and FN represent the corresponding samples. The confusion matrix [6, 24-26] of the classification result is shown in Table 2.

Table 2. Confusion matrix of the classification

Actual	Predicted Positive	Predicted Negative
Positive	TP	FN
Negative	FP	TN

ding to the confusion matrix above, the accuracy, precision, and recall can be defined. Accuracy is the correct proportion of all predictions and is defined as

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{1}$$

Another typical metric is the F1 score, which is the weighted average of the precision and recall, defined as

$$F_1 = \frac{2TP}{2TP + FP + FN} \tag{2}$$

The F1 score combines the results of precision and recall. When the F1 value is high, the classification model is ideal. The F1 score ranges from 0 to 1.

Performance Evaluation

MATLAB was used for machine learning the data. 75% of the data is used as the training set and the remaining 25% is used as the test set. Firstly, the training set is used for training to generate a fully grown decision tree. Next, this classification model is used to predict the remaining 25% of the data. Finally, the classification model is analyzed using the performance indicators mentioned in the previous section. The divided dataset is shown in Table 3.

Table 3. The divided dataset

Segmented dataset	Total size	Not registered	Registered
Full dataset	17652	7813	9839
Training set (75%)	13239	5901	7338
Test set (25%)	4413	1912	2501

Decision tree

The decision tree obtained after machine learning on the training set is very large, including 88 layers. Due to space constraints, the authors only show the upper layers of the original decision tree in Figure 2.

The decision tree is used to predict the test set, and the performance metric of the decision tree is shown in Tables 4 and 5.

Table 4. Confusion matrix of decision tree

Actual	Predicted Positive	Predicted Negative
Positive	1551(TP)	950(FN)
Negative	871(FP)	1041(TN)

Table 5. Performance metrics of decision tree

Recall	Precision	Accuracy	F1
62.02%	64.04%	58.74%	0.63

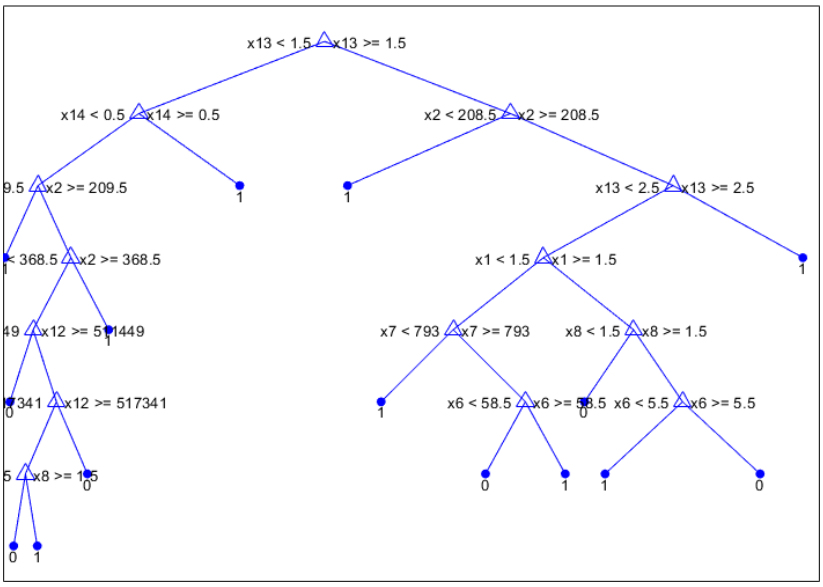

Figure 2. Part of the original tree

Random forest

In this paper, the software platform is Matlab, and use the software package RF_ Mexstandole-v0.02 to write the code of random forest.

The authors use 10, 50, 100, 200, and 500 trees to create the random forest respectively, and use these models to predict the test set. In order to avoid the influence of random factors, the authors calculate each model 10 times and finally take the average value as the final result. The experimental data are shown in Tables 6 and 7.

Table 6. Performance metric of different random forest

Number of trees	Recall	Precision	Accuracy	F1	Cost time (s)
10	71.11%	64.59%	62.82%	0.6769	0.6412
	70.04%	65.66%	63.12%	0.6778	0.5959
	71.35%	65.02%	62.60%	0.6804	0.5975
	70.16%	65.10%	62.86%	0.6753	0.6006
	71.10%	66.23%	63.54%	0.6858	0.6006
	69.74%	65.16%	62.51%	0.6737	0.6053
	71.13%	65.23%	62.31%	0.6805	0.5881
	70.48%	66.18%	63.44%	0.6825	0.6037
	69.16%	64.91%	62.11%	0.6697	0.5975
	69.44%	65.40%	62.59%	0.6736	0.5928
50	70.60%	66.95%	63.86%	0.6873	2.9890
	71.11%	67.21%	64.36%	0.6910	2.9578
	71.10%	67.05%	64.39%	0.6901	2.8938
	70.68%	66.11%	63.45%	0.6831	2.8954
	71.22%	65.55%	63.71%	0.6826	2.9157
	72.63%	65.33%	64.15%	0.6878	2.9125
	71.10%	65.61%	63.84%	0.6825	2.9016
	72.49%	66.98%	64.24%	0.6963	2.8969
	71.61%	67.64%	64.91%	0.6957	2.8985
	73.88%	66.30%	64.77%	0.6988	2.8876
100	73.60%	66.36%	64.83%	0.6979	5.8017
	71.90%	65.89%	63.86%	0.6877	5.8220
	68.36%	68.31%	64.05%	0.6833	5.7798
	70.46%	67.52%	64.84%	0.6896	5.7970
	70.14%	67.65%	64.05%	0.6887	5.8313
	73.05%	67.26%	65.09%	0.7004	5.8344
	69.69%	66.45%	63.59%	0.6803	5.7908
	71.61%	66.86%	64.39%	0.6915	5.8142
	70.24%	67.09%	64.23%	0.6863	5.8922
	71.61%	66.62%	63.61%	0.6903	5.7954
200	70.45%	67.42%	64.42%	0.6890	11.5519
	70.07%	66.37%	63.66%	0.6817	11.4926
	70.57%	67.11%	64.11%	0.6879	11.5254
	71.63%	67.46%	64.47%	0.6948	11.4567
	70.55%	68.12%	65.25%	0.6932	11.5909
	71.69%	67.79%	65.10%	0.6969	11.5316
	69.39%	69.16%	65.01%	0.6927	11.5566
	70.31%	69.03%	65.18%	0.6966	11.5410
	72.30%	65.97%	64.44%	0.6899	11.5176
	69.91%	67.57%	64.63%	0.6872	11.5472
500	70.80%	66.71%	63.95%	0.6869	28.7635
	71.91%	67.01%	64.64%	0.6937	28.7541
	70.33%	67.35%	64.61%	0.6881	28.7198
	72.03%	66.81%	65.07%	0.6932	28.5903
	72.67%	66.31%	64.98%	0.6934	28.7697
	71.76%	66.52%	64.56%	0.6904	28.8196
	70.99%	68.61%	65.39%	0.6978	28.8446
	69.99%	67.15%	63.91%	0.6854	28.6745
	71.29%	67.25%	64.38%	0.6921	28.6839
	69.05%	68.12%	64.07%	0.6858	28.8227

Table 7. Average value of different random forest

Number of trees	Recall	Precision	Accuracy	F1	Cost time (s)
10	70.37%	65.35%	62.79%	0.6776	0.6023
50	71.64%	66.47%	64.17%	0.6895	2.9149
100	71.06%	67.00%	64.25%	0.6896	5.8159
200	70.69%	67.60%	64.63%	0.6910	11.5312
500	71.08%	67.18%	64.56%	0.6907	28.7443

BP neural network

The authors created a neural network with three hidden layers, each containing 10 neurons, the number of iterations of the network is set to 1000, the training accuracy is 10-3, the learning rate is 0.01, and Maximum validation failures is 10 times. The training process is shown in Figures 3 and 4.

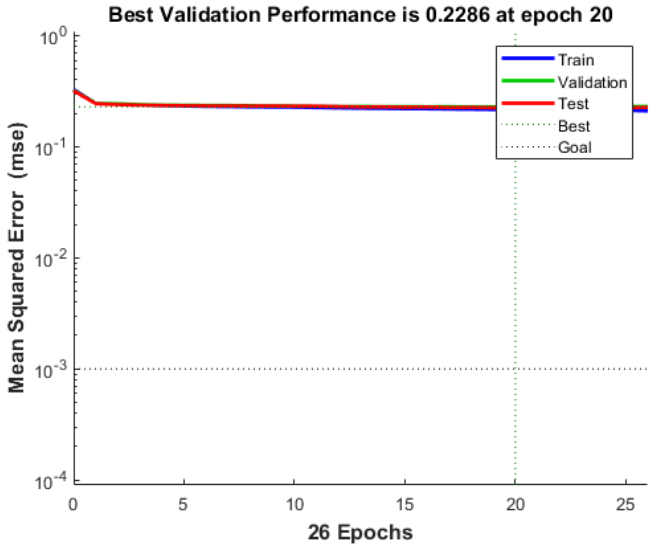

Figure 3. The performance of BP neural network

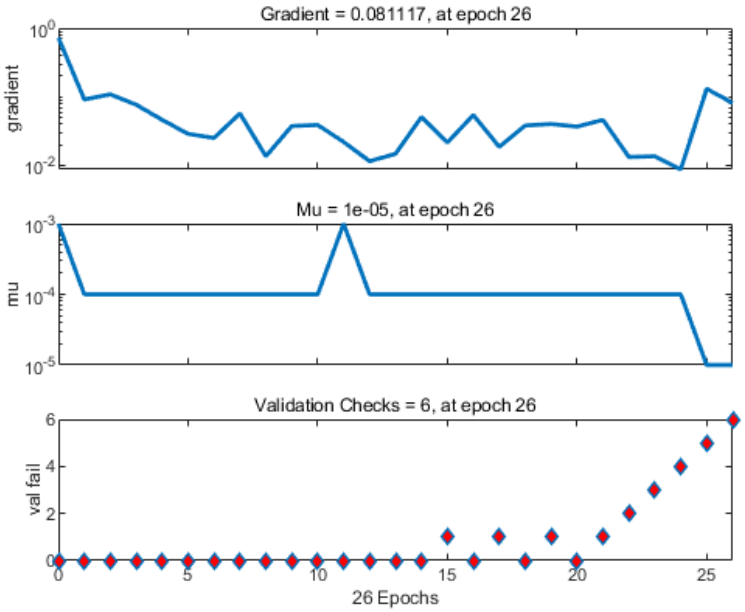

Figure 4. The training state of BP neural network

The BP neural network model is used to predict the test set. The final predicted performance indicators are shown in Table 8.

Table 8. Performance metric of BP neural network

Recall	Precision	Accuracy	F1
68.57%	64.68%	61.07%	0.6656

VOTING MODEL

In order to improve the performance of machine learning, the authors establish a voting model composed of multiple machine learning algorithms and use it to predict. The model consists of three algorithms mentioned above: decision tree, random forest, and BP neural network. The authors use hard voting and soft voting to predict the dataset respectively. Experiments verify the effectiveness of the scheme, and each performance index is significantly improved compared with a single algorithm.

For the same problem, different machine learning algorithms may give different prediction results. In this case, which algorithm is selected as the final result? At this time, the authors can concentrate a variety of algorithms to make different algorithms predict the same problem, and finally adopt the principle that the minority obeys the majority to select the final prediction result, this is the ensemble learning. It is also the idea of the voting model in this section. Ensemble learning [27, 28] is not an independent machine learning algorithm. It completes the learning task by constructing and combining multiple machine learners, that is, the researchers often say "absorbing the strengths of others". Ensemble learning can be used for classification problems, regression problems, feature selection, outlier detection, and so on. The random forest algorithm by the authors used earlier is a typical voting algorithm. Every tree in the forest will produce a prediction, and the prediction result with the most votes is the final prediction result.

The problem the authors study is the classification problem. Therefore, with the idea of ensemble learning, the authors construct a voting model to predict the enrollment of college freshmen. For this binary classification problem, the authors have three models: decision tree, random forest, and BP neural network. For each sample, the authors can get three prediction results, which are not necessarily the same. The authors choose the prediction result with the highest vote.

The specific voting process is to use the three models for machine learning and then summarize their results. There are two kinds of voting, hard voting, and soft voting. Hard voting is to choose the result with the largest number of votes as the final prediction result. Finally, a label is returned. Soft voting is to average the prediction probability obtained by each model and finally return a probability value.

In order to make the three algorithms consistent with the data set of the voting model (including the data set after random segmentation), the authors use orange 3 software to complete the experiment in this section, and their performance evaluation indicators are shown in Table 9.

Table 9. Performance metrics of different models

Model	Recall	Precision	Accuracy	F1
Random Forest	63.30%	63.00%	63.30%	0.631
Tree	61.40%	61.20%	61.40%	0.613
Neural Network	62.70%	62.40%	62.70%	0.624

The confusion matrices of the three models are shown in Tables 10, 11 and 12.

Table 10 Confusion matrix of decision tree

Actual	Predicted Positive	Predicted Negative
Positive	1666(TP)	802(FN)
Negative	902(FP)	1043(TN)

Table 11. Confusion matrix of random forest

Actual	Predicted Positive	Predicted Negative
Positive	1746(TP)	722(FN)
Negative	898(FP)	1047(TN)

Table 12. Confusion matrix of BP neural network

Actual	Predicted Positive	Predicted Negative
Positive	1734(TP)	734(FN)
Negative	913(FP)	1032(TN)

After summarizing the prediction results of the three models, their confusion matrix can also be obtained through hard voting and soft voting, as shown in Tables 13 and 14. Their performance evaluation indicators are shown in Table 15.

Table 13. Confusion matrix of hard voting

Actual	Predicted Positive	Predicted Negative
Positive	1746(TP)	722(FN)
Negative	882(FP)	1063(TN)

Table 14. Confusion matrix of soft voting

Actual	Predicted Positive	Predicted Negative
Positive	1753(TP)	715(FN)
Negative	844(FP)	1101(TN)

Table 15. Performance metrics of voting model

Model	Recall	Precision	Accuracy	F1
Hard voting	70.75%	66.44%	63.65%	0.69
Soft voting	71.03%	67.50%	64.67%	0.69

As can be seen from Table 15, the performance of the two voting methods is the same, the F1 score is the same. Compared with the single machine learning model, the performance of the voting model is significantly improved, as shown in Figure 5.

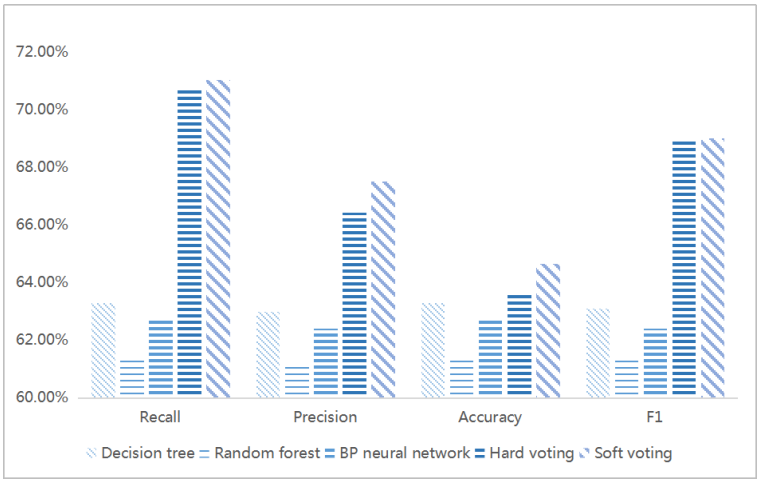

Figure 5. Compared with the single machine learning model

DISCUSSION

This research predicts the registration of college freshmen through machine learning algorithms, which can help colleges and universities make preparations in advance for various tasks to avoid chaos in the daily order of the university and damage to its reputation due to inadequate preparation. At the same time, the research results reveal the factors affecting the registration of freshmen, which has guiding significance for the enrollment and management of colleges and universities. However, the research has some shortcomings in terms of the limitations of data collection and processing, the specific implementation of the voting model, and the consideration of other potential factors. In the future, the authors should further improve the machine learning algorithms, conduct in-depth analysis by combining more data sources and features, apply the research results to more colleges and universities, in order to improve and expand this research.

Funding: This research was funded by the following programs: Research Capability Enhancement Project of Guangdong University of Science and Technology: Application Research of Artificial Intelligence Technology Based on Kunpeng Computing Platform; Natural Science Project of Guangdong University of Science and Technology(GKY-2022KYZDK-12, GKY-2022KYZDK-9, GKY-2022BSQD-39, GKY-2022BSQD-40); Innovation and School Strengthening Project of Guangdong University of Science and Technology(GKY-2022CQTD-2, GKY-2022CQTD-4, CQ2020062); Quality Engineering Project of Guangdong University of Science and Technology(GKZLGC2022018, GKZLGC2022271).

REFRENCES

[1] Z. C. Jia, Q. Y. Han, Y. Y. Li, Y. L. Yang, and X. Xing, "Prediction of Web Services Reliability Based on Decision Tree Classification Method," *Cmc-Computers Materials & Continua*, vol. 63, no. 3, pp. 1221-1235, 2020.

[2] K. D. Humbird, J. L. Peterson, and R. G. McClarren, "Deep Neural Network Initialization with Decision Trees," *IEEE Trans. Neural Netw. Learn. Syst.*, vol. 30, no. 5, pp. 1286-1295, 2019.

[3] X. Y. Mu, S. Remiszewski, M. Kon, A. Ergin, and M. Diem, "Optimizing decision tree structures for spectral histopathology (SHP)," *Analyst*, vol. 143, no. 24, pp. 5935-5939, 2018.

[4] M. T. Tran-Nguyen, L. D. Bui, and T. N. Do, "Decision trees using local support vector regression models for large datasets," *Journal of Information and Telecommunication*, vol. 4, no. 1, pp. 17-35, 2020.

[5] J. M. Sempere, "Modeling of Decision Trees Through P Systems," *New Gener. Comput*, vol. 37, no. 3, pp. 325-337, 2019.

[6] X. H. Chen, S. Y. Yu, Y. F. Zhang, F. F. Chu, and B. Sun, "Machine Learning Method for Continuous Noninvasive Blood Pressure Detection Based on Random Forest," *IEEE Access*, vol. 9, pp. 34112-34118, 2021.

[7] Q. F. Zhou, H. Zhou, and T. Li, "Cost-sensitive feature selection using random forest: Selecting low-cost subsets of informative features," *Knowledge-Based Systems*, vol. 95, pp. 1-11, 2016.

[8] Y. F. Li, F. X. Wu, and A. Ngom, "A review on machine learning principles for multi-view biological data integration," *Briefings in Bioinformatics*, vol. 19, no. 2, pp. 325-340, 2018.

[9] S. A. Naghibi, K. Ahmadi, and A. Daneshi, "Application of Support Vector Machine, Random Forest, and Genetic Algorithm Optimized Random Forest Models in Groundwater Potential Mapping," *Water Resources Management*, vol. 31, no. 9, pp. 2761-2775, 2017.

[10] J. C. Lyu and J. Zhang, "BP neural network prediction model for suicide attempt among Chinese rural residents," *Journal of Affective Disorders*, vol. 246, no. 1, pp. 465-473, 2019.

[11] Y. Zhang, Y. Zhu, X. Li, X. Wang, and X. Guo, "Anomaly Detection Based on Mining Six Local Data Features and BP Neural Network," *Symmetry*, vol. 11, no. 4, pp. 571, 2019.

[12] C.-M. Chen, B. Xiang, Y. Liu, and K.-H. Wang, "A secure authentication protocol for internet of vehicles," *IEEE Access*, vol. 7, pp. 12047-12057, 2019.

[13] D. Zhang, H. Dongru, L. Kang, and W. Zhang, "The generative adversarial networks and its application in machine vision," *Enterprise Information Systems*, pp. 1-21, 2019.

[14] L. Xiong, R.-S. Chen, X. Zhou, and C. Jing, "Multi-feature fusion and selection method for an improved particle swarm optimization," *Journal of Ambient Intelligence and Humanized Computing*, pp. 1-10, 2019.

[15] J. Wang, J. H. Qin, X. Y. Xiang, Y. Tan, and N. Pan, "CAPTCHA recognition based on deep convolutional neural network," *Math Biosci Eng*, vol. 16, no. 5, pp. 5851-5861, 2019.

[16] C. Cutter, "Amazon Wants to Train 29 Million People to Work in the Cloud: New programs seek to help people from Montana to Nigeria attain roles ranging from tech support to machine learning," *Wall Street Journal - Online Edition*, pp. N.PAG-N.PAG, 2020.

[17] P. Sejung and P. Han Woo, "A webometric network analysis of electronic word of mouth (eWOM) characteristics and machine learning approach to consumer comments during a crisis," *El Profesional de la Informacion*, vol. 29, no. 5, pp. 1-14, 2020.

[18] C. C. Zhang, S. Yu, G. J. Li, and Y. Xu, "The Recognition Method of MQAM Signals Based on BP Neural Network and Bird Swarm Algorithm," *IEEE Access*, vol. 9, pp. 36078-36086, 2021.

[19] L. Yang, L. Feng, L. Tian, and H. Dai, "Who Will Come: Predicting Freshman Registration Based on Decision Tree," *Computers, Materials \& Continua*, vol. 65, no. 2, pp. 1825--1836, 2020.

[20] E. V. A. Sylvester, P. Bentzen, I. R. Bradbury, M. Clement, J. Pearce *et al.*, "Applications of random forest feature selection for fine-scale genetic population assignment," *Evolutionary Applications*, vol. 11, no. 2, pp. 153-165, 2018.

[21] Y. Zhang, C. Zhang, Y. Zhao, and S. Gao, "Wind speed prediction with RBF neural network based on PCA and ICA," *Journal of Electrical Engineering*, vol. 69, no. 2, pp. 148-155, 2018.

[22] B. Kaminski, M. Jakubczyk, and P. Szufel, "A framework for sensitivity analysis of decision trees," *Cent. Europ. J. Oper. Res.*, vol. 26, no. 1, pp. 135-159, 2018.

[23] D. Hand and P. Christen, "A note on using the F-measure for evaluating record linkage algorithms," *Statistics and Computing*, vol. 28, no. 3, pp. 539-547, 2018.

[24] R. L. M. Robinson, A. Palczewska, J. Palczewski, and N. Kidley, "Comparison of the Predictive Performance and Interpretability of Random Forest and Linear Models on Benchmark Data Sets," *Journal of Chemical Information and Modeling*, vol. 57, no. 8, pp. 1773-1792, 2017.

[25] D. Yates and M. Z. Islam, "FastForest: Increasing random forest processing speed while maintaining accuracy," *Inf. Sci.*, vol. 557, pp. 130-152, 2021.

[26] B. Yu, H. Z. Wang, W. X. Shan, and B. Z. Yao, "Prediction of Bus Travel Time Using Random Forests Based on Near Neighbors," *Computer-Aided Civil and Infrastructure Engineering*, vol. 33, no. 4, pp. 333-350, 2018.

[27] Y. Guo, X. Wang, P. Xiao, and X. Xu, "An ensemble learning framework for convolutional neural network based on multiple classifiers," *Soft Comput*, vol. 24, no. 5, pp. 3727-3735, 2020.

[28] H. Y. Su and C. R. Huang, "Enhanced Wind Generation Forecast Using Robust Ensemble Learning," *IEEE Transactions on Smart Grid*, vol. PP, no. 99, pp. 1-1, 2020.