Using the data of students’ learning process recorded in the network teaching platform to predict students’ learning performance, assist teachers to analyze learning situation, formulate teaching strategies, and warn about students’ learning state is a hot spot in the field of mixed curriculum research in recent years. In view of the complexity, heterogeneity, and security of college educational administration data and the difficulty of predicting and analyzing college students’ achievements, this paper designs a college educational administration management system platform based on improved random forest algorithm. Combining the advantages of three data-driven prediction algorithms, namely, random forest, extreme gradient boosting (XGBoost), and gradient boosting decision tree (GBDT), a model based on improved random forest algorithm is proposed. It is proved that this method is a noninferior prediction method. Secondly, the model is applied to practical problems to solve the problem of predicting college students’ grades. An experiment is carried out on the real data set provided by a municipal education bureau. The results show that the proposed model not only achieves good prediction accuracy, but also solves the stability problem of the model after adding new data, which will contribute to the iterative optimization of the model, improve the universality of the model, and help continuously track the learning behavior characteristics of college students in different semesters.

1. Introduction

The issue of higher education has been a hot social issue for many years, from "elite education" to "mass education" [1]. Colleges and universities are faced with problems such as uneven quality of students and severe employment situation as they expand enrolment scale. Not only is student achievement an important index to evaluate the teaching quality of colleges and universities, but it is also closely related to student management and employment guidance. As the society attaches more and more importance to higher education, how to improve teaching quality through scientific prediction and analysis of college students’ scores has become the focus of the education department [2]. The traditional method is to use the knowledge of mathematical statistics to set the score line to classify the scores of college students, but it is difficult to find the potential factors that affect the scores [3].

Wang et al. [4] analyzed the variation trend of college students’ scores and urban-rural differences based on the score data of 985 college students, and the analysis results have certain practical literature value. However, this literature is a descriptive study, which only analyzes the law of the development of college students’ scores but does not provide an effective prediction model of scores, so it cannot be applied to the prediction of individual scores of college students.

Cole et al. [5] proposed to explore the strong correlation between college subjects based on frequent pattern and predict students’ scores of several courses in the future. This method has two limitations. First, the strong pattern correlation method determines that the score of a course can only be determined by the current few scores or even one score, which limits the improvement of its prediction accuracy. Second, the prediction method divides the scores into four grades, which cannot accurately and quantitatively predict the scores.
Rastrollo-Guerrero et al. [6] applied genetic neural network to the analysis of college students’ scores and achieved accurate score prediction. However, the experiment of this method is not complete. Only 16 results of neural network training and prediction are obtained, the experimental results are not statistically credible, and there is not enough empirical analysis to prove the generalization ability of the model method.

In literature [7], decision tree (DT) is used to predict examinee scores. By calculating the information entropy and information gain of feature attributes, a decision tree with maximum gain rate is constructed to establish classification rules and analyze prediction models. However, the prediction accuracy of continuous fractions is poor, which may lead to overmatching and overfitting. Literature [8] adopted Support Vector Machine (SVM) to the prediction of the scores of relevant courses in colleges and universities. However, there are some shortcomings in data collection and mixed prediction. Literature [9] adopted naïve Bayesian model (N-B) to collect examinees’ characteristic attributes and calculate conditional probabilities under different categories to predict final performance, ultimately improving the accuracy of prediction. In real life, attributes that affect performance are often closely related. However, in the Bayesian model, all attributes are independent from each other, resulting in poor classification accuracy.

Foreign scholars have also carried out relevant research on this. Abu Saa et al. [10] selected 8 important attributes by calculating the information gain rate of each attribute characteristic from the 18 attributes that affect students’ performance and constructed a decision tree with the selected 8 important attributes to predict students’ performance. Naylor [11] studied the scores of 300 students in an Indian university and found that students’ scores were greatly affected by such factors as home address, annual family income, mother’s education, living habits, and students’ historical scores. Hung et al. [12] proposed that students’ sociodemographic characteristics (such as race, gender, and economic status) and academic characteristics (such as school type and school performance) are closely related to their academic performance.

Although the above work has a good performance, there are still two problems. (1) It only considers the current work which has the characteristics of the selected influence on student achievement and ignores the influence of the selected features. (2) The current work assumes that the key factors in the influence degree of all the students are the same, ignoring the students’ individual differences. In fact, the degree of influence of different factors on the same student’s performance is different, and the degree of influence of different students by the same factor is also different.

Random forest algorithm is an ensemble learning algorithm based on tree model, which trains multiple decision trees to obtain the final prediction result [13]. Random sampling of input features and samples during training increases randomness and avoids interference caused by highly correlated features in multiple training. Therefore, it has good generalization ability [14]. However, the traditional random forest algorithm does not distinguish the decision trees with different classification ability, resulting in the same voting ability of the decision tree with good classification ability and the decision tree with bad classification ability. Moreover, multiple decision trees need to be generated in the training process, resulting in a long running time of the algorithm. Shah et al. [15] proposed parallel research of text classification based on random forest. Undersampling of training samples is used to reduce the influence of unbalanced data on the stochastic forest algorithm and improve the classification accuracy of the algorithm. However, there is no improvement on the Spark level; only the parallelization features of Spark are applied. Wang Cheng et al. [16] eliminated those decision trees with weak classification ability by setting thresholds, so as to improve the random forest algorithm and conduct experiments in Spark single-machine mode. However, simply setting the threshold may lead to the elimination of some useful decision trees and affect the final result. Moreover, the experiment was only carried out in Spark single-machine mode, which did not meet the conditions of parallel operation of Spark. Lötsch et al. [17] studied the application of distributed parallel random forest algorithm in biomedicine and proposed a weighted voting method for random forest algorithm through out-of-bag testing. A method of vertical data partitioning is proposed to reduce the cost of data communication between Spark distributed computing nodes. However, the classification accuracy of out-of-bag test is not high, the method of vertical data partition is too complicated, and the implementation cost is too high.

With the increasing popularity of online teaching, it is necessary to construct and perfect the educational administration system platform of colleges and universities [18]. The smooth operation of the educational administration data management platform is related to the daily guarantee of teaching operation in colleges and universities. Students’ learning process data recorded in the online teaching platform can be used to predict students’ academic performance, assist teachers in analyzing their learning situation, and formulate teaching strategies.

With the acceleration of the informatization construction in higher vocational colleges, problems in the expansion and integration, energy consumption, and other aspects of the software and hardware resources of the educational administration management system platform have become increasingly prominent [19]. They are embodied in the low ecological sustainability of the system, the waste of server performance in daily operation, and the insufficient carrying capacity under the condition of high concurrency. Therefore, this paper designs a college educational administration management system platform which improves the random forest algorithm. In order to verify the effectiveness of the system, an experiment was carried out on a real data set provided by a municipal education bureau. The results show that the proposed model not only achieves good prediction accuracy, but also solves the problem of stability of the model after the addition of new data, which will contribute to the iterative optimization of the model, improve the universality of the model, and continuously track the
learning behavior characteristics of students in different semesters.

The main innovations of this paper are as follows:

1. A model combining the advantages of random forest, XGBoost, and GBDT is proposed, and the method is proved to be a noninferior prediction method.

2. By applying this method to practical problems, the prediction problem of college students’ scores can be solved so as to formulate corresponding teaching strategies in advance.

This paper consists of four main sections: Section 1 gives an introduction, Section 2 presents the methodology, Section 3 analyzes and discusses the results, and Section 4 concludes the paper.

2. Methodology

2.1. Improved Random Forest Algorithm. The principle of random forest algorithm is to use Bagging sampling method to randomly select certain data from the original data set and form a number of training subsets with release. Randomly selected feature attributes are used to train these subsets to obtain the corresponding decision trees, and then the test sets are imported into the decision trees to obtain the corresponding classification results. Finally, the final classification results are selected by voting.

In traditional random forest algorithms, the time of Bagging sampling increases exponentially with the size of the data set. And Spark allocates extra storage space to store the sampled training set during this process. At the same time, a large amount of disk data interaction is performed, which reduces the efficiency of parallelization of the algorithm. Therefore, this paper proposes a Data Index Sampling (DIS) table. After sampling, the index number of the sampled data is obtained and recorded in the DIS table without the need for real space allocation and data partition. The DIS representation is shown in Table 1.

The “randomness” of random forest algorithm is reflected not only in the formation of training data set by random sampling, but also in the construction of decision tree by random selection of feature attributes. For space allocation and Spark’s interaction with disk, we put all the feature names in an array in order. Then, the feature name is randomly selected, and the corresponding index number (array subscript) is put into a Random Feature Index (RFI) table. Then, RFI and DIS are assigned to slave computing nodes in the Spark cluster so that each computing node has one or more DIS tables and RFI tables.

1. **Create a DIS table to store the index number corresponding to the original data obtained during each sampling process.**

2. **Put all feature names in an array in sequence; then, extract a certain number of feature names each time in the form of array subscripts, and save them in the RFI table.**

3. **Allocate DIS tables and RFI tables to slave computing nodes in the Spark cluster so that each distributed computing node has one or more DIS tables and RFI tables.**

4. **Use the C4.5 algorithm to construct a decision tree on each distributed computing node and perform the calculation task of information gain ratio.**

5. **Summarize the intermediate results of each distributed computing node, and submit them to the master node for subsequent decision tree construction.**

6. **The parallelization process of SP-RF is shown in Figure 1, where DIS_RDD and RFI_RDD indicate that training samples and feature arrays are placed in two RDD storage units, respectively.**

2.2. Gradient Boosting Decision Tree Algorithm. GBDT is an iterative decision tree model proposed by Jerome Friedman [20]. It consists of several decision trees. Based on boosting idea in ensemble learning, n weak learners are finally combined into a strong one through multiple iterations. The specific training process is shown in Figure 2.

The algorithm steps are as follows:

1. **Initialize the weak learner:**

\[
F_0(i) = \arg \min_c \sum_{j=1}^{w} L(y_{ij}, c).
\]

2. **For the number of iterations \(n = 1, 2, \ldots, z\) and sample \(x = 1, \ldots, w,\) calculate the gradient:**

\[
a_{nx} = -\left( \frac{\partial L(y_{ix}, F(ix))}{\partial F(ix)} \right)_{F(i0)-F_{i}=0},
\]

3. **Sample fitting regression tree model is used to obtain the leaf node region \(R_{xy}\) corresponding to the \(z\)-th tree, where \(y = 1, 2, \ldots, y.\)**
2.3. Extreme Gradient Boosting Algorithm. XGBoost is a new boosting algorithm with gradient lifting thought in mind [21]. XGBoost algorithm obtains the optimal solution by generating the second-order Taylor expansion of the loss function and considering adding regular terms to the loss function. It can automatically call the CPU for multi-threaded parallel computation to achieve higher prediction accuracy. The specific training process of the algorithm is shown in Figure 3.

The algorithm steps are as follows:

(1) Calculate the objective function:

\[ Y(f_n) = \sum_{x=1}^{i} L\left(j_x, \hat{f}_x^{n-1} + f_n(i_x)\right) + \Omega(f_n), \]  

\[ \Omega(f_n) = \gamma N + \frac{1}{2} \lambda \|m\|^2, \]  

where \( j_x \) is the actual value; \( \hat{f}_x^{n-1} + f_n(i_x) \) is the model predicted value; \( N \) is the number of leaf nodes; and \( M \) is leaf node fraction.

(2) Calculate the second-order Taylor spread:

\[ f(i + \Delta i) \approx f(i) + f'(i)\Delta i + \frac{1}{2} f''(i)\Delta i^2, \]

\[ a_x = \frac{\partial L(j_x, \hat{f}_x^{n-1})}{\partial \hat{f}_x^{n-1}}, \]

\[ b_x = \frac{\partial^2 L(j_x, \hat{f}_x^{n-1})}{\partial \hat{f}_x^{n-2}}. \]

(3) Calculate the decision tree complexity formula:

\[ \Omega(f_n) = \gamma N + \lambda \frac{1}{2} \sum_{y=1}^{N} m_y^2. \]  

(4) Substitute (7), (8), and (9) into (6):

\[ Y(f_n) = \sum_{y=1}^{N} \left[ \left( \sum_{x \in X_y} a_x \right) m_y + \frac{1}{2} \left( \sum_{x \in X_y} b_x + \lambda \right) m_y^2 \right] + \gamma N. \]  

(5) Calculate (9) to obtain the optimal weight of leaf node \( m_y^* \) and update the objective function:

\[ m_y^* = \frac{\sum_{x \in X_y} a_x}{\sum_{x \in X_y} b_x + \lambda}, \]

\[ Y(f_n) = \frac{1}{\gamma} \sum_{y=1}^{N} \left( \sum_{x \in X_y} a_x \right)^2 + \gamma N. \]
The function \( \ln() \) increases monotonically in its domain \( R^+ \); logarithmic operations do not change the monotonicity of the original data. It can effectively compress data scale and reduce data variability.

2.4. Virtualization Platform Construction. System platform virtualization includes server virtualization, network virtualization, and storage virtualization. Among them, the original educational administration system server resources are integrated through the host layer, and network virtualization and storage virtualization are configured using existing hardware devices. The architecture of cloud computing educational administration system platform is shown in Figure 4.

Based on the analysis of the situation of current educational administration management platform, the following resources need to be further optimized and integrated:

(1) Educational administration management system client: the server is mainly used by teaching administrators, with low average utilization rate of CPU and memory resources, and prone to downtime in the case of concurrent student course selection.

(2) Database server: only the database software of the educational administration management system is deployed on the physical database server. The CPU and memory usage are low on ordinary days, but the performance requirements cannot be met even in the case of high concurrency when schools evaluate teaching or students select courses.

(3) Low-performance blade server: this type of server equipment is old, with low performance and high failure rate, such as the old educational system platform WEB server, client server, and database server.

In the early stage, host pool P01 is set up in the host layer of cloud computing educational management platform solution architecture, and hosts at the same network layer can be deployed in the P01 host pool in the later stage. Building a certain number of clusters can ensure that the educational administration platform works stably under high load. In this deployment study, dynamic resource scheduling and high availability functions of the cluster need to be set up. The number of VMs on a physical server must be within the upper limit. In this design, the platform of cloud computing educational administration system will integrate two high-performance servers with virtualization technology. It is planned that each server will be allocated five virtual CPUs, and the database and storage will be deployed on one high-performance physical server.

(1) CPU requirements. According to the virtual CPU requirements of the servers to be deployed in the educational affairs system, five virtual CPUs are allocated to each of the two high-performance servers, and a total of 10 virtual CPUs are calculated. Currently, the server CPUs of the educational affairs management system include Intel and AMD. Three newly purchased high-performance servers are configured with Intel CPUs, and the other two blade servers are configured with AMD CPUs. Both types of CPUs support multiple cores. A single core is equivalent to a physical CPU, in accordance with the principle of ensuring efficient and stable server operation. The CPU load of the physical machine should be controlled within 90%. According to the statistics on the resource usage of the educational administration system, the average CPU usage of the server is 20%–30%, and the reasonable average physical CPU usage should be calculated in the range of 60%–90%. The carrying capacity ratio of physical CPUs to virtual CPUs should be 1 : 3. Four physical CPUs are required for the virtual deployment service platform.

(2) Network adapter requirements. Construct the educational administration system platform based on cloud computing, and divide the network adapter of the server into management network adapter, service network adapter, and storage network adapter according to different functions. Each server must be
configured with at least one storage NIC and one service NIC to meet the requirements of virtual platforms. That is, each server must be configured with at least Gigabit NICs.

(3) Memory requirements. At present, the new version of educational administration management system, Linux operating system, and Oracle database are all 64-bit. 16 G RAM is planned for each virtual server to ensure the stable operation of all virtual servers under high load, that is, a total of 160 G RAM.

In this study, virtual switches vswitch0 and vswitch-web need to be configured to run management data and service data, respectively. The vswitch-web virtual switch requires a service NIC to be bound. The web storage data is achieved by configuring the vswitch-storage virtual switch to bind to a storage NIC.

To ensure smooth operation of cloud computing educational administration data management platform under high load, port aggregation (LACP) technology is used to ensure the availability and stability of the system when a single point of failure occurs. Based on the design of virtualization platform, IP SAN and FC SAN are configured for storage sharing in the teaching management system to achieve HA stability. The two physical network adapters of the storage device are bound as one logical network adapter bond0 to enhance IP SAN stability. An intelligent elastic architecture is constructed by combining physical ports of two switches into one virtual switch through IRF, enabling users to log in to the IRF system regardless of geographical locations and devices, facilitating systematic management of all devices in the virtual system.

3. Result Analysis and Discussion

3.1. Data Set Description. A real data set was provided by the education bureau of a city to carry out the experiment, which consists of the results of three mock exams of college students in the city in 2017 and their scores of relevant courses. In order to make the experimental results more accurate and reliable, it is necessary to preprocess the data set. Separate social sciences and computer science students, and remove records of college students with the same name from the same school. The final data set contained records of 10,138 computer science examinees and 4,874 social sciences examinees. Each examinee record contains 22 characteristic attribute values, including three mock test scores of college examinees and their school, family background, examinee category, and relevant course scores.

The experimental environment of this part is as follows: Python 3.7, JetBrains PyCharm 2018.1 × 64, Windows 7, Intel i7-4790 CPU @ 3.60 GHZ processor, 4 GB RAM.
3.2. Evaluation Indicators. In this paper, the prediction accuracy is evaluated by hit rate (HR) and compared with other methods. The hit ratio is defined as follows.

\[
BR = \frac{\sum_{p \in P}|R(p) \cap N(p)|}{\sum_{p \in P}|N(p)|},
\]

where \( P \) represents the set of examinees in the test set and \( p \) represents each examinee in \( P \). \( R(p) \) represents the relevant courses predicted by the characteristics of the training set, and \( N(p) \) represents the actual test results on the test set.

3.3. Comparison Algorithm. To verify the performance of this paper’s model in predicting scores in colleges and universities, it is compared with the following 6 methods.

1. Method 1 is according to the overall college-related course performance prediction results \( X \cdot O_p \).
2. In method 2, the Naive Bayes model is used to calculate the conditional probability of each characteristic attribute of the examinee’s \( p \) mock test score \( S_p \) in the relevant course results in colleges and universities, so as to predict \( p \) score \( \bar{O}_p \).
3. In method 3, the decision tree model is used to calculate the information entropy and information gain selection splitting attribute of each feature of the examinee’s \( p \) mock test score \( S_p \), and the decision tree with maximum gain rate is constructed to predict the relevant course score \( \bar{O}_p \) of \( p \) university.
4. Method 4 uses multilayer perceptron to extract the characteristic attributes of candidate \( p \) from the mock test score \( S_p \), so as to predict \( p \)’s college-related course score \( \bar{O}_p \).
5. Method 5 uses LSTM to capture the short-term characteristic \( S_t \) of candidate \( p \) from the input \( S_p \) with timing, so as to predict \( p \)’s college-related course score \( \bar{O}_p \).
6. Based on LSTM, method 6 adds forward and backward propagation in the hidden layer. The long-term dependence relationship between \( p \) scores was captured by two-way running forward and backward, so as to predict \( p \)’s college-related course scores \( \bar{O}_p \).

The network parameter settings of literature [22–26] and the proposed model are shown in Table 3. Meanwhile, in order to eliminate the randomness of the experimental results, two operations are performed in this paper. (1) In order to obtain more reliable prediction, the training set and test set were randomly divided into 9:1, 8:2, 7:3, and 6:4 for multiple experiments. (2) 10 experiments were performed on each data set, and their average value is taken as the final experimental result.

Tables 4 and 5 show the hit ratio of the six prediction methods under different data set partitioning proportions. It can be seen from the tables that the model proposed in this paper has the highest hit ratio. This shows that the model has the best prediction accuracy.

Figures 5 and 6 show the comparison results of hit ratio between the algorithm in this paper and other five algorithms. The following can be seen from Figures 5 and 6: (1) When the ratio of training set and test set is 9:1, the hit ratio of the six methods reaches the highest value, which means that key features can be learned when the useful information of training set is enough, thus improving the accuracy. (2) The prediction accuracy of literature [22] is generally high, because the neural network has strong nonlinear fitting ability and can capture deeper characteristics of the examinee. Therefore, the prediction accuracy can be greatly improved by using neural network. (3) Based on the model of literature [23], the hit rate is higher than those of other methods. This proves that the short-term characteristics of

| Table 3: Network parameter settings. |
|-------------------------------|-------------------------------|
| Literature [25] Social sciences | Number of layers: 3 layers. |
| Literature [22] Social sciences | Dimension: [64, 32, 16] |
| Literature [26] Social sciences | Number of layers: 3 layers. |
| Literature [23] Social sciences | Dimension: [128, 64, 32] |
| Literature [24] Social sciences | Output vector dimensions: 64 |
| Proposed Social sciences | Output vector dimensions: 128 |
| Computer | Social sciences | Long-term characteristic dimension: 128 |
| Proposed | Computer | Long-term characteristic dimension: 16 |

| Table 4: Prediction performance comparison (computer). |
|----------------|----------------|----------------|----------------|----------------|
| Computer | 9:1 | 8:2 | 7:3 | 6:4 |
| Literature [26] | 0.834 | 0.828 | 0.827 | 0.828 |
| Literature [22] | 0.867 | 0.858 | 0.863 | 0.866 |
| Literature [25] | 0.889 | 0.886 | 0.884 | 0.884 |
| Literature [23] | 0.913 | 0.890 | 0.897 | 0.898 |
| Literature [24] | 0.912 | 0.894 | 0.892 | 0.895 |
| Proposed | 0.915 | 0.897 | 0.900 | 0.901 |

| Table 5: Prediction performance comparison (social sciences). |
|----------------|----------------|----------------|----------------|----------------|
| Social sciences | 9:1 | 8:2 | 7:3 | 6:4 |
| Literature [26] | 0.836 | 0.814 | 0.819 | 0.823 |
| Literature [22] | 0.877 | 0.875 | 0.869 | 0.868 |
| Literature [25] | 0.900 | 0.874 | 0.886 | 0.866 |
| Literature [23] | 0.905 | 0.887 | 0.895 | 0.887 |
| Literature [24] | 0.894 | 0.886 | 0.889 | 0.891 |
| Proposed | 0.907 | 0.891 | 0.898 | 0.896 |
examinees have a certain degree of influence on the scores of relevant courses in colleges and universities. The prediction accuracy of the model of literature [24] is similar to that of literature [23], indicating that it is not significant to capture the characteristics of examinees in the prediction of scores.

4) The model in this paper used the model in literature [23] and the model in literature [24] to fully perceive the multiple characteristics of users and achieved the highest hit rate among the six methods.

4. Conclusion

The achievement of college students is the focus of current education departments. Scientific analysis of the achievement can help teachers and students arrange and adjust their learning plan reasonably, so as to improve the achievement of students. Traditional methods use the knowledge of statistics and data mining to discover the implicit relationship between achievements, but the accuracy of the methods is limited when faced with multiple nonlinear data. Therefore, this paper designs a college educational administration management system platform based on improved random forest algorithm. The performance of the model is evaluated on the real data set of the gap between college students in a city. It is proved that the prediction efficiency of the model is better than that of other comparison models, and it can accurately predict the future scores of college students and provide a reliable reference for modern student management based on data. The conclusion of this study is obtained from a mixed course of a university. Whether the accuracy of convergence prediction results can be obtained from a larger sample of students still needs to be further verified. In addition, this study only carried out experiments for students majoring in social sciences and computer science, and the scores of students majoring in art and sports will be predicted and analyzed in the future.

Data Availability

The labeled data set used to support the findings of this study is available from the author upon request.

Conflicts of Interest

The author declares no conflicts of interest.

Acknowledgments

This study was supported by Jiaxing Nanyang Polytechnic Institute.

References

in predicting the student graduation timeliness,” Journal Online Informatika, vol. 6, no. 1, p. 11, 2021.