

Research Article

Predicting Learning Behavior Using Log Data in Blended Teaching

Shu-Tong Xie ^{1,2}, Zong-Bao He,¹ Qiong Chen ³, Rong-Xin Chen ¹, Qing-Zhao Kong,⁴
and Cun-Ying Song⁵

¹School of Computer Engineering, Jimei University, Xiamen 361021, China

²Digital Fujian Big Data Modeling and Intelligent Computing Institute, Jimei University, Xiamen 361021, China

³College of Computer and Information Sciences, Fujian Agriculture and Forestry University, Fuzhou 350002, China

⁴School of Science, Jimei University, Xiamen 361021, China

⁵School of Foreign Languages, Jimei University, Xiamen 361021, China

Correspondence should be addressed to Qiong Chen; qchen@fafu.edu.cn

Received 10 May 2021; Revised 15 June 2021; Accepted 14 August 2021; Published 24 August 2021

Academic Editor: Jiwei Huang

Copyright © 2021 Shu-Tong Xie et al. This is an open access article distributed under the Creative Commons Attribution License, which permits unrestricted use, distribution, and reproduction in any medium, provided the original work is properly cited.

Online and offline blended teaching mode, the future trend of higher education, has recently been widely used in colleges around the globe. In the article, we conducted a study on students' learning behavior analysis and student performance prediction based on the data about students' behavior logs in three consecutive years of blended teaching in a college's "Java Language Programming" course. Firstly, the data from diverse platforms such as MOOC, Rain Classroom, PTA, and cnBlog are integrated and preprocessed. Secondly, a novel multiclass classification framework, combining the genetic algorithm (GA) and the error correcting output codes (ECOC) method, is developed to predict the grade levels of students. In the framework, GA is designed to realize both the feature selection and binary classifier selection to fit the ECOC models. Finally, key factors affecting grades are identified in line with the optimal subset of features selected by GA, which can be analyzed for teaching significance. The results show that the multiclass classification algorithm designed in this article can effectively predict grades compared with other algorithms. In addition, the selected subset of features corresponding to learning behaviors is pedagogically instructive.

1. Introduction

With the rapid advancement of Internet technology, blended teaching mode has emerged in recent years. Different from the previous online teaching mode, blended teaching includes an integration of learners' online learning with traditional face-to-face classroom learning, which not only can guarantee the degree of interaction between teachers and students but also can ensure the efficiency of students' online learning owing to the supervision of teachers. Through analyzing students' online learning behavior and classroom performance, teachers can gain insight into the comprehensive information of students, making it possible to tailor their teaching to students' needs to further improve the teaching quality. Therefore, there is a need to study how to fully exploit the data of students' online and offline learning behavior to effectively guide students' individualized learning.

At present, most of the research work is typically based on students' learning behavior data to predict whether students can pass examinations, obtain certificates, or dropout of classes, etc., which are essentially binary class problems. Besides, much of the work is done to extract coarse-grained features from data for prediction. In this work, however, we particularly focus on the prediction of students' performance in four categories (excellent, good, passed, and failed) upon their learning behavior data. Specifically, we add a genetic algorithm- (GA-) based feature selection and binary classifier selection to the error correct output codes (ECOC) framework to filter the learning behavior features of the original data, in order to improve the accuracy of multiclassification in many ways. According to the fine-grained results of feature selection and correlation analysis, the learning features that have a more significant impact on students' final grades are selected from their

learning behaviors for analysis, which is conducive to teachers' individualized teaching.

The structure of this article is organized as follows. Section 2 discusses the related work. Section 3 primarily gives a description of the course and data involved. Section 4 briefly introduces ECOC algorithm and GA as well as details the design and implementation of the ECOC-GA algorithm. Section 5 presents the experimental setup and the comparison combined with analysis of the experimental results and then puts forward some implications for teaching. We conclude the whole work in final section.

2. Literature Review

In this section, we will briefly review the research and application of machine learning and mining technology in online learning. Ang et al. conducted a comprehensive survey of the research related to big educational data that has emerged in the past few years [1]. They reviewed the overview and classification of big educational data research, data collection and mining efforts for big educational data, methodological and technical issues involved in big educational data, and challenges faced by big educational data. Related research studies showed that prediction of learning effectiveness had an important research value on the education field, thereby receiving much attention. Basing on a dataset collected from XuetangX, one of the largest MOOCs from China, Qiu et al. proposed a latent dynamic factor graph (LadFG) to predict students' homework completion and whether they can successfully pass the examinations and obtain the certificates [2]. By analyzing students' activity logs on the MOOCs, Xu and Yang summarized various learning motivations of students and designed a classification algorithm based on support vector machine (SVM) to predict whether students can obtain a certificate [3]. Working with multisource heterogeneous data from two courses by the Peking University on Coursera, *Data Structures and Algorithms and Introduction to Computing*, Zhang et al. analyzed students' learning content, identified important concepts in the course, assessed students' knowledge states through the quiz, and designed algorithms to predict student dropout in MOOCs [4]. Yu et al. identified seven types of cognitive participation models of learners based on their video clickstream records and built practical machine learning models using K-nearest neighbors (KNN), SVM, and artificial neural networks (ANN) to make predictions about whether students can pass the course examinations [5]. In order to significantly improve the teaching efficiency of traditional courses and online MOOCs, Meier et al. used history of teaching data of the course (mainly including assignments, quizzes, midterm examination, etc.) to predict the likely performance (good/poor) of students in subsequent courses so that teachers can initiate interventions promptly for students performing poorly [6]. Ulloa-Cazarez et al. proposed a genetic programming (GP) algorithm to predict whether students can pass the final examination [7]. Trying to identify students who have difficulty in course learning, Xu et al. developed a machine learning algorithm with a bilayered structure consisting of multiple base

predictors and a cascade of ensemble predictors [8]. This algorithm could be used to make predictions based on students' progressive performance states. Additionally, a clustering method based on latent factor models and probabilistic matrix factorization was proposed to discover the correlation among the courses to further construct more effective base predictors. The results suggested that the method had a positive impact on assessing student performance and was able to provide implications of teaching intervention for instructors. Hussain et al. used machine learning algorithms including ANN, SVM, Logistic Regression (LR), Naive Bayes (NB), and Decision Trees (DT) to analyze data recorded by the TEL system. The model was trained according to the data of student performance from the previous week and then tested on the data from the following week [9]. Hence, the differences in performance among various algorithms could be compared. The results showed that ANN and SVM achieved higher accuracy than other algorithms. Overall, the method employed was highly real time and especially helpful in allowing teachers to identify underprepared students and facilitate interventions before the next week's class.

In addition to the research on the prediction of learning effectiveness, there is also an increasing body of work in the field of students' behavioral patterns and the influencing factors of learning. Basing on the Spark platform, Hu et al. proposed a method for analyzing students' video watching behavior in MOOCs and validated it with the data of the cauX platform [10]. The experimental results indicated that the method could quickly and accurately analyze the characteristics of video watching behavior, which is conducive to explaining the time-distributed video watching behavior in MOOCs. Besides, it could also guide instructors to determine a reasonable video length to draw more students' attention and reduce their dropout rate. Theoretically guided by Moore's theory of transactional distance, Hew et al. adopted approaches incorporating supervised learning, sentiment analysis, and hierarchical linear modeling to analyze the cognitive features of 6,393 students in 249 random MOOCs, with MOOC learners' satisfaction as a measure of success [11]. The results revealed that teacher, content, assessment, and schedule had significant effects on student satisfaction. Onan et al. developed a document-clustering model based on weighted word embedding to identify question topics on posts of MOOC forum [12]. Later, Onan designed a long short-term memory networks (LSTM) to classify the sentiment for about 70,000 MOOC reviews, which can achieve a high classification accuracy [13]. In addition, sentiment analysis on both Twitter data and students' evaluation data are also investigated recently [14–16]. Due to teaching inertia and difficulty of fixed curriculum setting in coping with the IT field where knowledge is updated rapidly, Chen et al. proposed a data-based framework to evaluate the relationship between taking courses and getting employment through the DT expression [17]. Based on it, a computer course group recommendation algorithm was proposed to provide a basis for optimal course configuration. The results showed that the course optimization and recommendation results based on this method had a significant contribution to student

employment. Yousef and Sumner conducted a comprehensive analysis of more than 200 articles on MOOC research and discussed the main research findings and research directions on MOOC in recent years around the issues of classification of MOOC, MOOC learners, communication between MOOC providers and participants, and assessment for students' learning behavior [18]. Mubarak et al. used a twofold analysis method combining visual and predictive analyses to make an initial prediction of learners' performance through the analysis results of data visualization and then built the RNN-LSTM model for learners' behavior in video clickstreams to improve the predictability of learner performance [19]. Zhang et al. constructed a SPOC-based flipped classroom teaching model by integrating the OBE concept, and the results confirmed that the model in small classes had equally ideal effects on both teachers and students [20]. Chen et al. introduced and evaluated an interactive visual analytics system called ViSeq, showing that it could visualize the learning sequences of different learner groups to aid users in exploring them of MOOCs at multiple levels of granularity and comprehending the underlying patterns behind the learning sequences [21]. Combining TAM with TPB and collecting data from questionnaires completed by university learners, Wang et al. constructed a theoretical model of MOOC learning performance mechanisms [22]. The analysis of learning groups' characteristics revealed that factors such as perceived usefulness, learning attitudes, subjective norms, and perceived behavioral control had disproportionate impacts on learning performance, while perceived ease of use was not a crucial factor. The results provided a scientific basis for MOOC instructional design and instructional resources allocation. Cobos and Ruiz-Garcia proposed a learning intervention system called edX-LIS for providing MOOC learners with information about their progress regularly and making suggestions by their performance [23]. Contrast experiments had shown that the intervention strategies provided by this system had a positive impact on learners' motivation, persistence, and engagement. García-Molina et al. proposed an algorithm for automatic scoring of learners' performance in MOOC forums [24]. Simultaneously, they collected data for an exploratory study that proposed three schemes with different input parameters. The results showed a moderate positive correlation between the forum grades provided by the algorithm and the grades obtained through the summative assessment activity of MOOC. Xie conducted a survival analysis for the video viewing duration of MOOCs and developed a mathematical model to understand the evolutionary mechanisms underlying the characteristics of students' video viewing behavior [25]. The study also explored the potential role of memory in the complexity of learning behaviors. Wen et al. conducted a detailed analysis of learning behavior patterns of MOOC learners and proposed a novel simple feature matrix for maintaining the local correlation information of learning behavior [26]. The study found that learners often exhibited similar learning behaviors over consecutive days. By making use of it, a brand new convolutional neural network (CNN) model was also proposed to improve the accuracy of dropout prediction. Yu et al. applied the software product line (SPL) in software engineering

techniques to a framework for data analysis, which made the process of data analysis reusable [27]. A practical machine learning model was constructed on the basis of this framework to predict learning outcomes through students' learning behavior. The framework retained data collection, data cleaning, feature extraction, and model prediction as core components. As a result, a complete and reusable learning behavior analysis system came into being.

3. Data Description

3.1. Curriculum Setting. Our algorithm was applied to data collected from a blended course of *Java Language Programming* in a college for three consecutive years. The course aimed at understanding the basic knowledge of Java and learning Java programming techniques, with a total of 68 class hours. While participating in traditional courses in classroom, students also studied online through e-learning platforms, such as Rain Classroom, MOOC, PTA, and Blog. They would have a final examination at the end of the semester.

3.2. Data Description. In this article, we collected learning data generated by students in three grades of a college while participating in the course (the three different grades are represented by grade 1, grade 2, and grade 3). Among them, 94, 127, and 130 students in grades 1, 2, and 3 participated in the course, respectively. The data were mainly obtained from four platforms: MOOC platform, Rain Classroom, Programming Teaching Assistant (PTA), and cnBlog. The details are summarized in Table 1.

3.2.1. Data of MOOC. The data of MOOC platform mainly includes 102 lecture videos, 6 peer grading assignments, 8 chapter quizzes, forum discussions, online examination, and other types of learning activity data. The specific fields are shown in Table 2.

3.2.2. Data of Rain Classroom. The Rain Classroom data mainly contains two categories: preclass preparation and classroom participation. 25 preclass preparation cases and 9 preclass preparation cases are included in the Rain Classroom data of grade 1; 27 preclass preparation cases and 16 classroom participation cases are included in the Rain Classroom data of grade 2; 26 preclass preparation cases and 15 classroom participation cases are included in the Rain Classroom data of grade 3. The detailed descriptions are listed in Table 3.

3.2.3. Data Integration. In addition to the data from MOOC platform and Rain Classroom platform, cnBlog assignments and PTA examinations are also set up to enrich the course content. Therefore, it is necessary to integrate MOOC data, Rain Classroom data, cnBlog data, and PTA data. The integrated data are merged and aligned with the data of students' final examination grades, which are classified into four levels: excellent, good, passed, and failed.

TABLE 1: Course data source.

Semester	Number of students	Data source
Fall semester 2017-2018 (grade 1)	94	Rain Classroom and cnBlog
Fall semester 2018-2019 (grade 2)	127	Rain Classroom and PTA
Fall semester 2019-2020 (grade 3)	130	MOOC, Rain Classroom, and PTA

TABLE 2: Data of MOOC.

Features	Explanation	Note
Student information	It is made up of student number, name, class, etc.	
Task response	Number of tasks completed/task completion percentage	
Number of discussions	Forum discussion	
Chapter learning times		Students can watch lectures multiple times
Video viewing duration		Duration (10.3 minutes)
SPOC-debugging.mp4		Duration (1.7 minutes)
...	There are 102 videos here	.
Introduction.mp4		Duration (1.7 minutes)
Number of posts/reply discussions		
Suggested score	Student scores suggested by the system	
Course video progress	Number of completed course videos/total number	
Chapter test progress		
Total performance	Students' online scores	Hundred mark
Five level (ABCDE)	Turn grades into levels	
Certificate issuing status		Yes/no
Peer assessment—1		Score
Submission of peer graded assignment	There are 6 peer grading	Yes/no
Participation in peer graded assignment	Assignments here	Yes/no
...		
Assignment grade of unit 1		Hundred-mark
Assignment submission time of unit 1	Time: YYYY-MM-DD hh:mm:ss	
...		
Assignment grade of unit 8	Details of the assignments and	
Assignment submission time of unit 8	Quizzes for a total of eight units	

TABLE 3: Data of Rain Classroom.

Features	Explanation	Note
Students' ID/name/number	Students' information	
Total score		
Attendance rate	Attendance times/total times	
Number of notices read		
Total pages viewed PPT		Preclass preparation
Total duration	Time for watching courseware	Preclass preparation
Completion time	Time to complete courseware	Preclass preparation
Score	Answer situation in courseware	Preclass preparation
Sign-in method		Classroom participation
Number of pop-ups	Total number of pop-ups sent	Classroom participation
Cumulative score		Classroom participation

Finally, after data filtering, feature transformation, and other preprocessing operations, the grade 1 dataset contains 82 samples in 138 dimensions; the grade 2 dataset includes 118 samples in 151 dimensions; the grade 3 dataset comprises 115 samples in 291 dimensions.

4. The Design of ECOC-GA

Classification is a common problem in machine learning, which can be further divided into binary classification and

multiclassification. In this work, the ECOC algorithm is utilized to integrate multiple binary classifiers to solve the multiclassification problem.

4.1. Introduction to ECOC. ECOC first came from the field of communication as a technique for correcting information transmission errors in networks. In the field of machine learning, ECOC algorithm essentially serves as an ensemble learning framework, using different base classifiers to learn

prior knowledge at different levels on dataset. Accordingly, the overall performance of models can be effectively improved for the mutual complementation of the classifiers [28–30].

ECOC algorithm mainly consists of three basic steps: the encoding step, the training step, and the decoding step [31]. The encoding strategies are mainly divided into data-independent encoding and data-dependent encoding. To be specific, data-independent encoding can be further divided into One-vs-All (OVA), One-vs-One (OVO) [32], Dense Random (DR) [33], and Sparse Random (SR) [34], whereas data-dependent encoding includes D-ECOC, ECOC-ONE, Forest-ECOC, and other types [35]. An ECOC algorithm guides the training and prediction of multiple binary classification models by generating a code matrix. Take the encoding process of the OVA matrix as an example, the matrix row vector represents a class and the column vector represents a binary classifier. As depicted in Figure 1, the first column of the code matrix $\{1, -1, -1, -1\}$ indicates the first column corresponding to the binary classifier h_1 treats the data of class $\{C_1\}$ as a positive group and the data of other classes as a negative group, and then the model h_1 can be trained utilizing the training data divided into two classes.

In the decoding stage, when predicting the output label of a certain sample, the prediction results returned by each dichotomizer will form a result vector. By calculating the distance between this result vector and the code word in the code matrix, the code word with the smallest distance between them is selected as the final prediction result.

4.2. Feature Selection Algorithm. The quality of data is the key to machine learning. If the original dataset has too many redundant features, it will not only cause serious interference to the whole learning process but also cause a waste of computation time and memory. Feature selection can select some of the most important features in the dataset, which enhances models' generalization ability and reduces overfitting, as well as improves the interpretability between features and prediction targets.

GA is a global algorithm to search the solution space and find the optimal solution. It is formed by simulating Darwin's natural selection theory and combining it with the theory of biological evolution in genetics [36]. Selection, crossover, and mutation constitute the genetic operator of GA. In addition, setting of the initial population, development of the individual coding strategy, design of the fitness function, design of the genetic operator, and setting of the control parameters form the core of GA. Related studies have proved that GA as a feature selection algorithm [37, 38] can select more relevant features in comparison with traditional algorithms, thus further improving the predictive performance of models.

4.3. Multiclassification Algorithm Based on ECOC and GA. Classical ECOC algorithms take all features as input variables and adopt the same kind of binary classifiers set to fit models and make predictions. In contrast, we designed a multiclassification algorithm based on ECOC and GA for

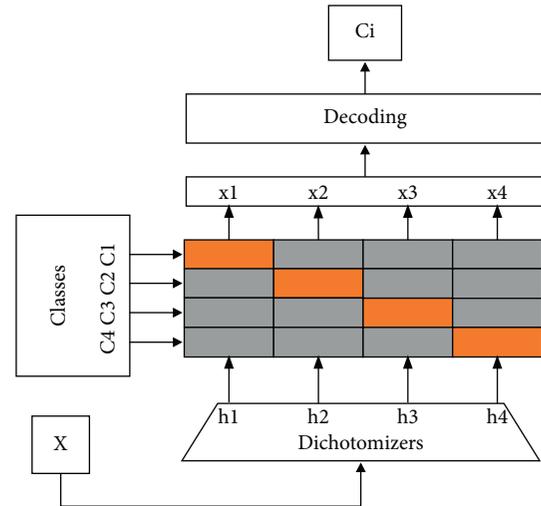


FIGURE 1: OVA code matrix for a four-class problem.

performance prediction, which uses a heterogeneous set of binary classifiers. GA is used not only for feature selection but also for binary classifier selection for each column in the code matrix. The specific steps of the algorithm are as follows:

- (1) Representation of chromosome: feature subset coding is the mapping of sample feature selection in the dataset. Hence, each feature corresponds to a binary number (1 represents selection of the feature, 0 represents non-selection of the feature). As a result, we can get n -bit binary chromosome codes (n is the dimension of training dataset). The base classifier sequence coding is the mapping of the selection of binary classifiers in each column of the code matrix. There are three optional base classifiers, and accordingly, each column corresponds to one trinary number and m -bit trinary chromosome codes appear (m is the number of columns in code matrix).
- (2) Setting of relevant parameters: initialization settings for various parameters such as the number of iterations of GA, population size, elite preservation rate, random selection rate, and mutation rate.
- (3) Population initialization: the chromosomes of each individual are coded by a random function. Each individual has two chromosomes. According to the chromosome coding sequence, some students' learning behaviors and the optimal set of base classifiers are all selected.
- (4) Selection: the selection operation of GA is divided into two main parts. Firstly, the top 20% of the most adapted individuals in current population are selected based on the elite preservation rate. Then, some of the less adaptive but surviving individuals are selected among the remaining individuals based on the random selection rate. The method is consistent with the randomness of natural selection, and it also allows the whole population to obtain a higher fitness value.

- (5) Crossover: we adopt the method of single-point crossover, randomly selecting two individuals from the population as parents and randomly selecting a position of the individual's chromosome as a crossover site. Using the crossover site as the intercept point, we intercept two parts of the parents, respectively, to reassemble two new individuals. Assuming a parent individual's chromosome is 1000,0010,0010,1010, the other is 0101,1010,0001,1110, and the crossover site is 2. Then, the two offspring individuals' chromosomes after crossover are 1100,0010,0010,1010 and 0001,1010,0001,1110, respectively. The population is replenished to keep the population size constant during the iteration.
- (6) Mutation: according to the mutation rate, one of the chromosomes of an individual is randomly selected for gene mutation to obtain a new individual and update the population.
- (7) Termination condition: the algorithm can be stopped when the number of iterations meets the requirement, and the result can be used as the current optimal solution.

We use an ECOC-based framework for models' training and prediction and adopt the 5-fold cross-validation scheme to evaluate the performances. The average value of the accuracy is calculated as the fitness to measure how much GA improves the performance of the overall multiclassification problem. The feature subset and base classifiers sequence with high fitness are selected to enter the next iteration, and the final individual with the highest fitness, i.e., containing the optimal feature subset and the optimal set of base classifiers, is obtained. The flowchart of the proposed ECOC-based algorithm (referred to as ECOC-GA) is shown in Figure 2.

5. Experimental Results and Analysis

5.1. The Setting of Experiments. The parameter of ECOC algorithm: encoding methods are OVO, DR, and DECOG; base classifiers are LR, SVM, and Bayes; decoding method is soft decoding; the rest are set with default parameters.

The parameter of GA: the number of individuals in the population is 85, the random selection rate is 0.5; the mutation rate of the feature subset is 0.03; the mutation rate of the base classifiers sequence is 0.08; the elite preservation rate is 0.2, and the maximum number of iterations is 65. In addition, Random Forest and XGBoost are provided by the scikit-learn library with default parameter settings.

In this work, all experiments are conducted using the 5-fold cross-validation scheme, and the mean value of accuracy is used as an evaluation index for the classification performance and fitness values in GA.

5.2. Comparison of Experimental Results

5.2.1. Comparison with Traditional Ensemble Learning Algorithms. There are many classical ensemble learning

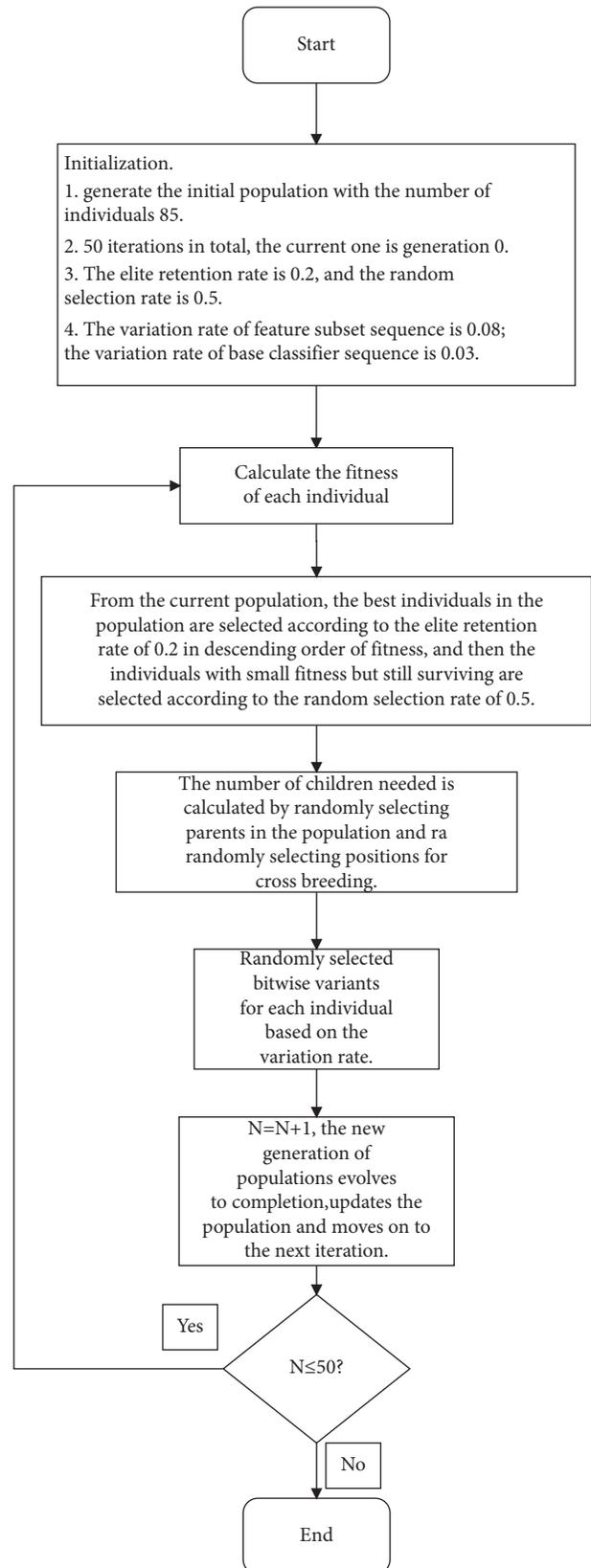


FIGURE 2: Flowchart of ECOC-GA algorithm.

algorithms in machine learning such as Random Forest and XGBoost. The comparison of the ECOC-GA ensemble

learning algorithm and the two traditional ensemble learning algorithms is illustrated in Figure 3. It can be found that the classification accuracy of Random Forest and XGBoost is lower, and the ECOC-GA algorithm has a significant improvement in accuracy for performance prediction compared with them, indicating that the ECOC-GA algorithm outperforms traditional ensemble learning algorithms.

5.2.2. Comparison with Other ECOC Algorithms. On the basis of ECOC, ECOC-GA algorithm achieves better prediction performance using the feature selection and base classifier selection through GA. The experimental results between the two kinds of algorithms are summarized in Tables 4–6.

As shown in Table 4, when OVO coding method is adopted in the experiments: The prediction accuracy of the ECOC-GA on the grade 1 dataset is improved by about 7.4%, 6.2%, and 11.7% over ECOC (SVM), ECOC (Bayes), and ECOC (LR), respectively; the prediction accuracy of the ECOC-GA on the grade 2 dataset is increased by about 1.4%, 12.5%, and 1.3% over ECOC (SVM), ECOC (Bayes), and ECOC (LR), respectively; the prediction accuracy of the ECOC-GA on the grade 3 dataset is lifted by about 1.7%, 3.5%, and 0.4% over ECOC (SVM), ECOC (Bayes), and ECOC (LR), respectively. Therefore, the proposed algorithm has significantly improved the prediction accuracy over classical ECOC algorithms on all three datasets.

When DR coding method is used, we can see from Table 5 that the prediction accuracy of the ECOC-GA on the grade 1 dataset is improved by about 0.7%, 3.7%, and 9.9%, respectively, compared with ECOC (SVM), ECOC (Bayes), and ECOC (LR). Compared with ECOC (SVM), ECOC (Bayes), and ECOC (LR), the prediction accuracy of ECOC-GA on grade 2 dataset is increased by about 1.8%, 3.5%, and 1.3%, respectively. Compared with ECOC (SVM), ECOC (Bayes), and ECOC (LR), the prediction accuracy of ECOC-GA on grade 3 dataset is lifted by about 1.7%, 5.9%, and 4.8%, respectively. Thus, using DR coding method, the ECOC-GA also has a small performance improvement in three datasets.

Similar results are also reflected in the experiments using DECOC coding method. As demonstrated in Table 6, when DECOC coding method is applied, the prediction accuracy of ECOC-GA on the grade 1 dataset grows by about 3.9%, 1.5%, and 8.9%, respectively, compared with ECOC (SVM), ECOC (Bayes), and ECOC (LR). Compared with ECOC (SVM), ECOC (Bayes), and ECOC (LR), the prediction accuracy of ECOC-GA on grade 2 dataset is improved by 3.0%, 11.6%, and 3.9%, respectively. Compared with ECOC (SVM), ECOC (Bayes), and ECOC (LR), the prediction accuracy of ECOC-GA on grade 3 dataset is increased by about 0.5%, 5.3%, and 3.1%, respectively.

In addition, we also observe that the algorithms have slightly lower accuracy on the grade 3 dataset. The possible reason is that the grade 3 dataset has more dimensions of features, and the corresponding solution space is larger than the other two datasets. Hence, it is more difficult to search

for an optimized parameter, and the classification accuracy is reduced. However, on the whole, the prediction accuracy of ECOC-GA improved to different degrees over ECOC algorithms on all three datasets.

In summary, even with different datasets or coding methods, the prediction accuracy of the ECOC-GA is consistently higher than that of other ECOC algorithms. Therefore, the ECOC algorithm with GA for feature selection and base classifiers selection has a superior performance.

5.2.3. Comparison of Different Methods at Different Time Stages. Furthermore, in order to find out students' learning effectiveness at different learning stages so that teachers can intervene effectively in advance, we use the ECOC-GA method to predict students' learning performance at different time stages. Subsequently, we compare the performance of algorithms at different learning stages. Specifically, from the end of midterm examination to the end of the course, every two weeks is considered as a learning stage, and a total of six stages can be formed. Experiments are conducted using ECOC, ECOC-GA, Random Forest, and XGBoost, respectively.

For different datasets, different base classifiers, and different coding methods, there are obvious differences in prediction accuracy between algorithms based on ECOC, i.e., ECOC and ECOC-GA, and classical ensemble learning algorithms, i.e., XGBoost and Random Forest. Because the ECOC-GA uses multiple base classifiers, it is not fair to compare it directly with other ECOC algorithms. Therefore, we select the best results among three ECOC algorithms based on single-classifier and compare them with the ECOC-GA.

When OVO is used, the results can be observed from Figure 4. With 18 cases on different datasets, the ECOC-GA algorithm can all obtain better results. Figure 5 shows that there are 18 cases in the datasets at different time stages when DR is used. The ECOC-GA can achieve 17 better results, 15 better results, and 18 better results in the datasets of grades 1, 2, and 3 at different time stages, respectively. Finally, as can be observed from Figure 6, there are also 18 cases on different datasets when DECOC is used. The ECOC-GA wins 15, 15, and 16 cases out of 18 cases on different time stage datasets at grades 1, 2, and 3, respectively.

In general, although the ECOC-GA does not always achieve better results than ECOC algorithms, it generally outperforms ECOC algorithms and has stronger robustness. In addition, XGBoost and Random Forest are less effective in predicting results at different time stages. These results fully demonstrate the good classification performance of the proposed ECOC-GA on the learning behaviors datasets.

5.2.4. Comparison of Time Complexity. Table 7 provides the running time of ECOC-GA. On the three datasets, both Random Forest and XGBoost can complete the training within 20 seconds, thus having the lowest complexity. ECOC algorithms can complete the training within 50 seconds regardless of different coding methods and different base

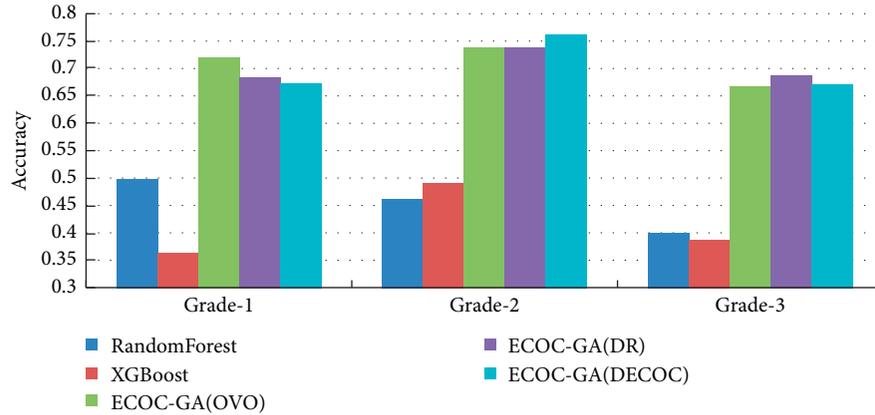


FIGURE 3: Average accuracy of ECOC-GA and traditional ensemble learning algorithms.

TABLE 4: Average accuracy of ECOC-GA and ECOC with OVO coding method.

Datasets	ECOC (SVM)	ECOC (Bayes)	ECOC (LR)	ECOC-GA
Grade 1	0.646 ± 0.043	0.658 ± 0.063	0.603 ± 0.043	0.720 ± 0.082
Grade 2	0.722 ± 0.045	0.611 ± 0.034	0.723 ± 0.037	0.736 ± 0.030
Grade 3	0.648 ± 0.044	0.630 ± 0.048	0.661 ± 0.054	0.665 ± 0.021

TABLE 5: Average accuracy of ECOC-GA and ECOC with DR coding method.

Datasets	ECOC (SVM)	ECOC (Bayes)	ECOC (LR)	ECOC-GA
Grade 1	0.676 ± 0.049	0.646 ± 0.062	0.584 ± 0.059	0.683 ± 0.055
Grade 2	0.717 ± 0.030	0.700 ± 0.058	0.722 ± 0.031	0.735 ± 0.029
Grade 3	0.670 ± 0.039	0.628 ± 0.048	0.639 ± 0.038	0.687 ± 0.029

TABLE 6: Average accuracy of ECOC-GA and ECOC with DECOC coding method.

Datasets	ECOC (SVM)	ECOC (Bayes)	ECOC (LR)	ECOC-GA
Grade 1	0.634 ± 0.043	0.658 ± 0.053	0.584 ± 0.063	0.673 ± 0.049
Grade 2	0.731 ± 0.028	0.645 ± 0.034	0.722 ± 0.034	0.761 ± 0.035
Grade 3	0.665 ± 0.038	0.617 ± 0.040	0.639 ± 0.029	0.670 ± 0.046

classifiers, so the complexity of the algorithms is also low. On the contrary, the ECOC-GA requires more than 3000 seconds to fit models. Hence, the complexity is relatively high.

Because the number of samples and features of the three datasets is different, the running time of the methods are also different. Grade 3 dataset has 290 dimensional features; the solution space available for searching is larger, and the computing time is longer. Grade 1 dataset has 137 features, and grade 2 dataset has 150 features. The computing time of them is similar, and both shorter than that of grade 3 dataset.

On the other hand, for the same dataset, the time complexity of different algorithms is not much different. Nevertheless, the average operation time using DR is slightly larger than OVO, whereas the average operation time using DECOC is the shortest. The reason is that when employing DR coding method to generate random matrix, if there is any illegal coding matrix, it should be deleted and new coding matrix is needed. However, the matrix generated by OVO is fixed as well as simple, and therefore, the complexity of the algorithm is smaller.

5.3. Analysis of the Educational Significance of Key Features. By calculating the Pearson correlation coefficient between students' learning behaviors and students' final grades, the

influence of different features on students' final grades can be observed.

5.3.1. Correlation Analysis of Features. Correlation analysis is performed to identify the features that are significantly correlated with grades. The Pearson correlation coefficient of each feature and final grade can be obtained. For the grade 1 dataset, by filtering the features with high Pearson correlation coefficients, we find high linear correlations between the total scores, attendance rates, cnBlog assignment scores, completion of some courseware, the usual performance, and the final grades for Rain Classroom. For the dataset of grade 2, it is still found that these features have a high linear correlation with final grades through filtering. The proportion of features with high correlation to the total features is high as well, which indicates that students in this grade may have higher utilization of e-learning platform. For the grade 3 dataset, through screening, we find that in addition to the above features, the chapter quizzes and final grades in MOOCs have a high correlation, and the rest of the MOOC features have a low correlation. The dimensions of dataset in grade 3 are high, but there are few features with high correlation with final

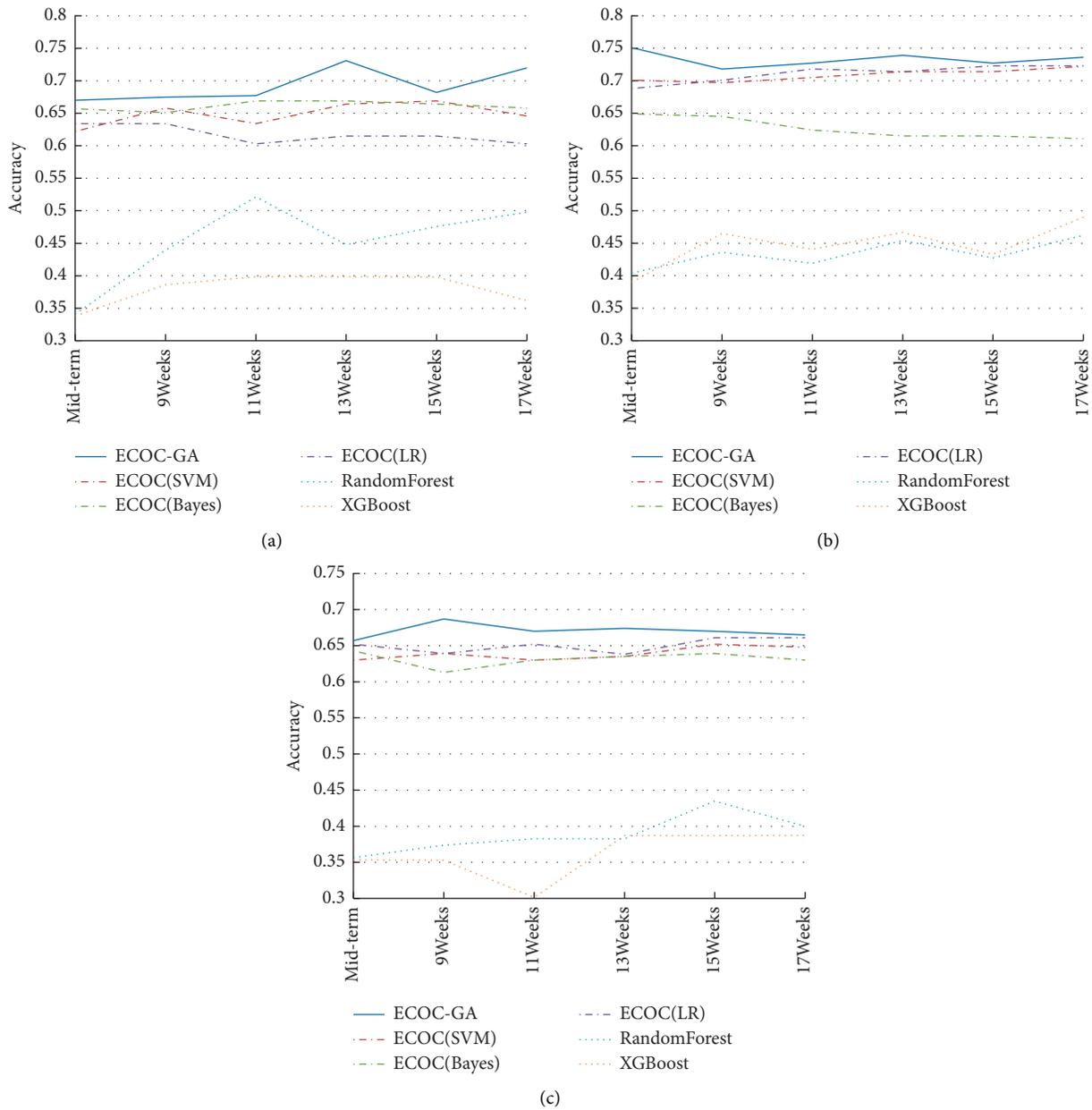


FIGURE 4: Accuracy comparison of different algorithms with OVO coding method. (a) Grade 1 dataset. (b) Grade 2 dataset. (c) Grade 3 dataset.

grades, which indicates that students in this grade may not pay much attention to e-learning, and their overall low final grades may also confirm this conjecture.

5.3.2. *Analysis of the Optimal Feature Subset and Its Educational Implications.* Using GA as the feature selection algorithm, the feature selection result corresponding to the final generation is the most optimal feature subset, and the accuracy of using this feature subset is also the highest. Therefore, these features are likely to have specific pedagogical significance. The ECOC-GA designed in this study has good prediction accuracy, so we decide to utilize the common features selected by the ECOC-GA based on the

three coding methods as the final result. Subsequently, we will analyze these features to extract useful information.

Due to the high dimensionality of the data, the number of retained features is still high after performing feature selection, although nearly half of the features are filtered. Thus, it is decided to combine the results of the correlation analysis and use the intersection of the features selected by the two methods to further cut down the number of features and select the features that have a higher degree of influence on the final grades. It is beneficial to analyze the pedagogical implications of the selected features. Specifically, for grade 1 dataset, 23 of the 42 features with high correlation are also selected by feature selection; for grade 2 dataset, 32 of the 60 features are selected; for grade 3 dataset, 18 of the 32 features

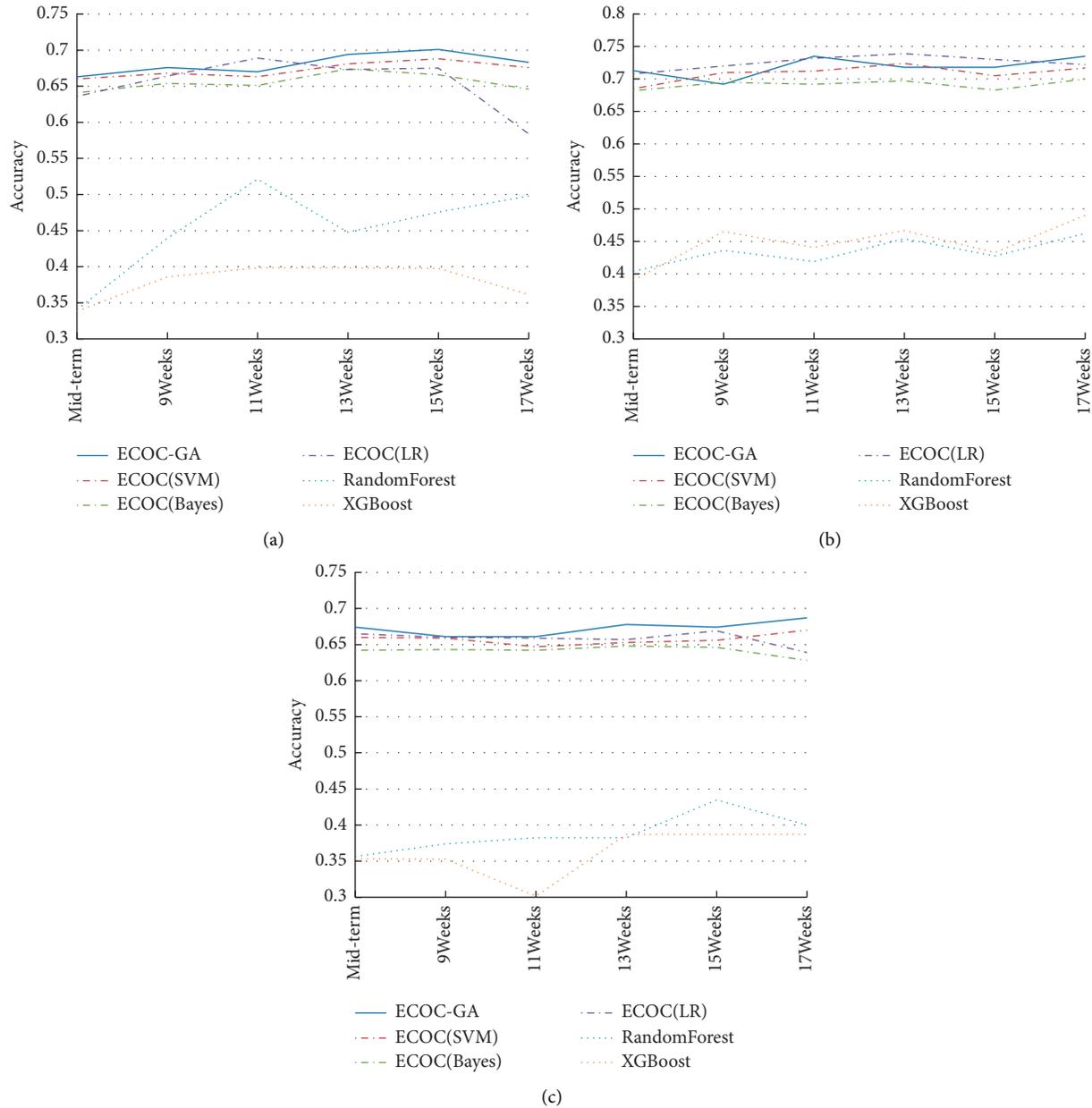


FIGURE 5: Accuracy comparison of different algorithms with DR coding method. (a) Grade 1 dataset. (b) Grade 2 dataset. (c) Grade 3 dataset.

are selected. There are 18 features among the 32 features with high correlation. The results also show that there is a corroborating relationship between the results of the correlation analysis and the results of the feature selection to some extent. The results are displayed in Table 8, where “courseware” indicates students’ prereading of the current content, “class status” indicates students’ learning performance in the current course content, and “xxx.mp4” refers to the progress and number of times students watched the video.

As listed in Table 8, for the students of grade 1, the learning contents of Java Language Programming such as Inheritance, Polymorphism, Interfaces, Nested classes, Collections, Object-Oriented, Exceptions, Files and Databases, and Multithreading have a significant impact on the

final grades. In addition, the completion of students’ cnBlog assignments also has an important impact on the final grades. For the students of grade 2, Object-Oriented, Inheritance, Polymorphism, Interfaces, Nested classes, Collections, Exceptions, Multithreading, and I/O flow have a high correlation with the final grade. For the students of grade 3, the main features that show high correlation are analogous to the aforementioned important features in terms of watching videos and previewing courseware alike. From the intersection of features of the above three datasets, it can be observed that Collections, Object-Oriented, Inheritance, Polymorphism, Interfaces, Nested classes, and Exceptions are the key and difficult points of the whole course. Hence, good command of relevant knowledge contributes greatly to students’ final grades. The correctness

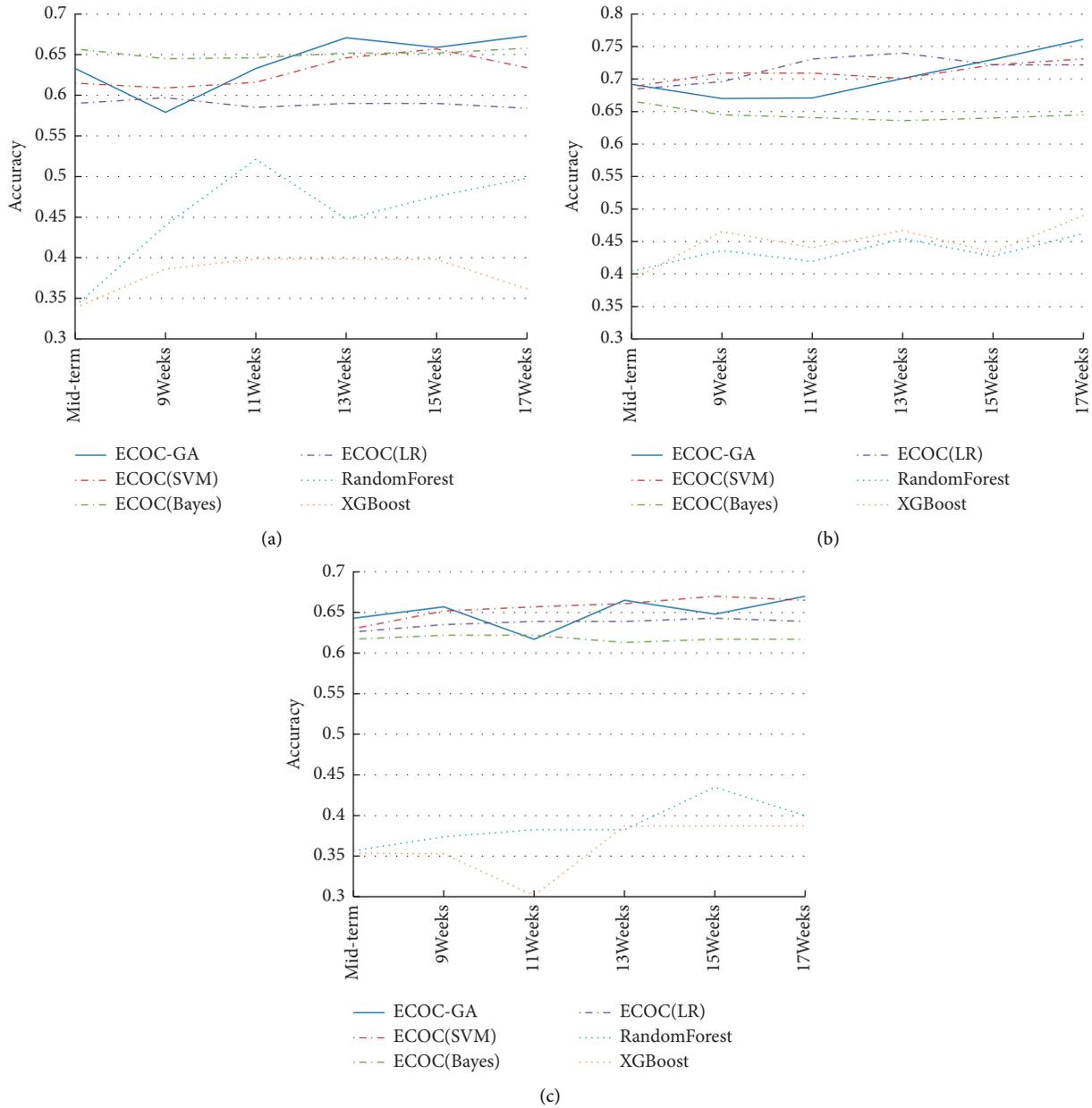


FIGURE 6: Accuracy comparison of different algorithms with DECOC coding method. (a) Grade 1 dataset. (b) Grade 2 dataset. (c) Grade 3 dataset.

TABLE 7: The running time of ECOC-GA (in seconds).

Datasets	ECOC-GA (OVO)	ECOC-GA (DR)	ECOC-GA (DECOC)	ECOC	XGBoost	Random Forest
Grade 1	3250	3332	3158	≈50	≈20	≈20
Grade 2	3544	3608	3392	≈50	≈20	≈20
Grade 3	5535	5808	5472	≈50	≈20	≈20

of the inference is further confirmed by the content selected in several assignments in the cnBlog.

The selected features can be organized into three major categories of learning behaviors: preview before class, course learning, and quizzes after class. To be concrete, the preview before class is divided into PPT viewing and precourse practice; the course learning is divided into videos watching and in-class practice; the quizzes after class is divided into

after-class homework and chapter quizzes. From the experimental results, it is clear that the pages of PPT viewed before class, the scores of in-class quizzes, and the completion of homework are more helpful to improve the final grades. As to the reason for this phenomenon, we believe that the more PPT pages students viewed before class, in other words, the greater his efforts and investment put in the current content, the better learning outcomes they will get in return. Besides,

TABLE 8: Feature selection results.

Datasets	Selected features
Grade 1	<i>Total score</i> , attendance rate, Java data types, and reference types—courseware, object-oriented programming—courseware, <i>inheritance and polymorphism</i> —courseware, interfaces and nested classes—courseware, GUI and swing—courseware, <i>Java collection framework and object oriented</i> —courseware, <i>exceptions</i> —courseware, multithreading—class status, networking—courseware, file and database—courseware, JSP and servlet—courseware, exception handling—class status, cnBlog (2nd, 3rd, 9th, 12th, 13th, and 15th assignment), <i>usual performance</i>
Grade 2	<i>Total score</i> , number of reading announcements, procedural oriented programming—courseware, <i>objects and classes</i> —courseware, <i>inheritance and polymorphism</i> —courseware, <i>interfaces and nested classes</i> —courseware, Java swing Java’s GUI—courseware, <i>Java collection framework</i> —courseware, <i>Java exceptions</i> —courseware, exception mechanism—class status, Java multithreading—courseware, Java input-output flow—courseware, database—class status, servlet and JSP—courseware, PTA midterm examination scores, <i>usual performance</i>
Grade 3	<i>Collections.mp4</i> , <i>Exception_Exception_Handling.mp4</i> , chapter quiz progress, collections—peer assessment, exceptions—chapter quiz, object oriented—chapter quiz, <i>interfaces and nested</i> —chapter quiz, collections—chapter quiz, <i>objects and classes</i> —courseware, exception mechanisms—class status, <i>inheritance and polymorphism</i> —courseware, I/O streams and file operations—class status, Java—network programming—courseware, PTA midterm examination scores, <i>usual performance</i>

the interspersed exercises assist students to verify their mastery of the content during the learning process. By doing this, students can consolidate what they have learned, and their sense of achievement can be stimulated greatly so that more efficient learning can be achieved, which can further create a virtuous circle. Meanwhile, the completion of after-lesson assignments reflects the students’ learning attitudes to some extent, so students who do well in cnBlog assignments must take every aspect of their online course learning seriously, which determines their higher final grades.

Another interesting phenomenon is that very few features of MOOC platform are selected, suggesting that students’ use of MOOC for e-learning does not correlate well with their final grades. Through communication with course instructors, it is found that MOOCs are only used as a teaching aid, so students may not pay much attention to it or have much investment. The viewing of more than 100 videos is also not very relevant to the final grades, probably because students are only “task-oriented” on the course in order to complete the task, which naturally does not reflect the students’ knowledge mastery. In addition, the PTA midterm examination results did not have a close relationship with the final grades, which is also worth considering and needs to be further explored.

In summary, the selected features have significant pedagogical significance and can aid teachers to fully understand the arrangement of teaching content and to suit the remedy to the case in future teaching by modifying the teaching content and curriculum appropriately as well as make full use of blended teaching methods to improve teaching effectiveness.

6. Conclusion

In this article, we propose a novel ECOC based on GA, which is applied to the mining of blended teaching data. The main findings are as follows:

- (1) It is experimentally demonstrated that the ECOC-GA algorithm has a large improvement in accuracy of prediction when compared with classical ECOC algorithms and the traditional ensemble learning algorithm, such as XGBoost and Random Forest. In

addition, the prediction performance of ECOC-GA is also better than ECOC algorithms, XGBoost, and Random Forest algorithm in general for prediction at different periods.

- (2) Through the combinations of feature selection and correlation analysis, the selected feature subsets are analyzed for their pedagogical significance, and the key and difficult points of the “*Java Language Programming*” course are found. On the basis of them, teachers can improve the teaching setup, enhance teaching quality, and help students consolidate key points and break through difficult points, so that students can effectively improve their performance. At the same time, we find that the cnBlog assignments are essentially helpful, which can be promoted and further enriched in future teaching. Conversely, MOOCs and PTA are not effective enough in practice. Accordingly, teachers should have second thoughts about the use and setting of MOOCs and PTA.

Future work can be improved in the following ways. Firstly, improve data quality and collect more teaching-related data. Secondly, improve the ECOC algorithm to make it more suitable for fitting smaller samples of higher dimensional data, thus further improving the prediction accuracy. Finally, GA can not only be used for feature selection but also can be combined with ECOC algorithm. So future research can explore the better ways of their combination.

Data Availability

The data used to support the findings of this study are available from the corresponding author upon request.

Conflicts of Interest

The authors declare that they have no conflicts of interest.

Acknowledgments

This work was supported in part by the Natural Science Foundation of Fujian Province of China (nos. 2020J01697,

2020J01707, 2020R0066, and 2018J01538), the Scientific Research Program of Fujian Bureau of Education, China (no. JAT200266), and the Opening Fund of Digital Fujian Big Data Modeling and Intelligent Computing Institute.

References

- [1] K. L.-M. Ang, F. L. Ge, and K. P. Seng, "Big educational data & analytics: survey, architecture and challenges," *IEEE Access*, vol. 8, pp. 116392–116414, 2020.
- [2] J. Qiu, J. Tang, T. X. Liu et al., "Modeling and predicting learning behavior in MOOCs," in *WSDM '16: Proceedings of the Ninth ACM International Conference on Web Search and Data Mining*, pp. 93–102, San Francisco, CA, USA, February 2016.
- [3] B. Xu and D. Yang, "Motivation classification and grade prediction for MOOCs learners," *Computational Intelligence and Neuroscience*, vol. 2016, Article ID 2174613, 7 pages, 2016.
- [4] M. Zhang, J. Zhu, Z. Wang, and Y. Chen, "Providing personalized learning guidance in MOOCs by multi-source data analysis," *World Wide Web*, vol. 22, no. 3, pp. 1189–1219, 2019.
- [5] C.-H. Yu, J. Wu, and A.-C. Liu, "Predicting learning outcomes with MOOC clickstreams," *Education Sciences*, vol. 9, no. 2, p. 104, 2019.
- [6] Y. Meier, J. Xu, O. Atan, and M. Van der Schaar, "Predicting grades," *IEEE Transactions on Signal Processing*, vol. 64, no. 4, pp. 959–972, 2015.
- [7] R. L. Ulloa-Cazarez, C. López-Martín, A. Abran, and C. Yáñez-Márquez, "Prediction of online students performance by means of genetic programming," *Applied Artificial Intelligence*, vol. 32, no. 9-10, pp. 858–881, 2018.
- [8] J. Xu, K. H. Moon, and M. Van Der Schaar, "A machine learning approach for tracking and predicting student performance in degree programs," *IEEE Journal of Selected Topics in Signal Processing*, vol. 11, no. 5, pp. 742–753, 2017.
- [9] M. Hussain, W. Zhu, W. Zhang, S. M. R. Abidi, and S. Ali, "Using machine learning to predict student difficulties from learning session data," *Artificial Intelligence Review*, vol. 52, no. 1, pp. 381–407, 2018.
- [10] H. Hu, G. Zhang, W. Gao, and M. Wang, "Big data analytics for MOOC video watching behavior based on Spark," *Neural Computing & Applications*, vol. 32, no. 11, pp. 6481–6489, 2019.
- [11] K. F. Hew, X. Hu, C. Qiao, and Y. Tang, "What predicts student satisfaction with MOOCs: a gradient boosting trees supervised machine learning and sentiment analysis approach," *Computers & Education*, vol. 145, Article ID 103724, 2020.
- [12] A. Onan and M. A. Toçoğlu, "Weighted word embeddings and clustering-based identification of question topics in MOOC discussion forum posts," *Computer Applications in Engineering Education*, vol. 29, no. 4, pp. 675–689, 2020.
- [13] A. Onan, "sentiment analysis on massive open online course evaluations: A text mining and deep learning approach," *Computer Applications in Engineering Education*, vol. 29, no. 3, pp. 572–589, 2020.
- [14] O. Aytuğ, "Sentiment analysis on Twitter based on ensemble of psychological and linguistic feature sets," *Balkan Journal of Electrical and Computer Engineering*, vol. 6, no. 2, pp. 69–77, 2018.
- [15] A. Onan, "Mining opinions from instructor evaluation reviews: a deep learning approach," *Computer Applications in Engineering Education*, vol. 28, no. 1, pp. 117–138, 2020.
- [16] A. Onan, "Sentiment analysis on product reviews based on weighted word embeddings and deep neural networks," *Concurrency and Computation: Practice and Experience*, Article ID e5909, 2020.
- [17] X. Chen, J. Zheng, Y. Du, and M. Tang, "Intelligent course plan recommendation for higher education: a framework of decision tree," *Discrete Dynamics in Nature and Society*, vol. 2020, pp. 1–11, 2020.
- [18] A. M. F. Yousef and T. Sumner, "reflections on the last decade of MOOC research," *Computer Applications in Engineering Education*, vol. 29, no. 4, pp. 648–665, 2020.
- [19] A. A. Mubarak, H. Cao, W. Zhang, and W. Zhang, "Visual analytics of video-clickstream data and prediction of learners' performance using deep learning models in MOOCs' courses," *Computer Applications in Engineering Education*, vol. 29, no. 4, pp. 710–732, 2020.
- [20] L. Zhang, Y. Xuan, and H. Zhang, "Construction and application of SPOC-based flipped classroom teaching mode in Installation Engineering Costcurriculum based on OBE concept," *Computer Applications in Engineering Education*, vol. 28, no. 6, pp. 1503–1519, 2020.
- [21] Q. Chen, X. Yue, X. Plantaz et al., "Visual analytics of learning sequence in massive open online courses," *IEEE Transactions on Visualization and Computer Graphics*, vol. 26, no. 3, pp. 1622–1636, 2018.
- [22] Y. Wang, C. Dong, and X. Zhang, "Improving MOOC learning performance in China: an analysis of factors from the TAM and TPB," *Computer Applications in Engineering Education*, vol. 28, no. 6, pp. 1421–1433, 2020.
- [23] R. Cobos and J. C. Ruiz-García, "Improving learner engagement in MOOCs using a learning intervention system: A research study in engineering education," *Computer Applications in Engineering Education*, vol. 29, no. 4, pp. 733–749, 2020.
- [24] S. García-Molina, C. Alario-Hoyos, P. M. Moreno-Marcos, P. J. Muñoz-Merino, I. Estévez-Ayres, and C. Delgado Kloos, "An algorithm and a tool for the automatic grading of MOOC learners from their contributions in the discussion forum," *Applied Sciences*, vol. 11, no. 1, p. 95, 2021.
- [25] Z. Xie, "Modelling the dropout patterns of MOOC learners," *Tsinghua Science and Technology*, vol. 25, no. 3, pp. 313–324, 2020.
- [26] Y. Wen, Y. Tian, B. Wen, Q. Zhou, G. Cai, and S. Liu, "Consideration of the local correlation of learning behaviors to predict dropouts from MOOCs," *Tsinghua Science and Technology*, vol. 25, no. 3, pp. 336–347, 2020.
- [27] C.-H. Yu, J. Wu, M.-C. Liu, and A.-C. Liu, "Adopting software product lines to implement an efficient learning analytics framework in MOOCs," *Journal of Information Science and Engineering*, vol. 37, no. 1, pp. 139–155, 2021.
- [28] M. Sun, K. Liu, Q. Wu, Q. Hong, B. Wang, and H. Zhang, "A novel ECOC algorithm for multiclass microarray data classification based on data complexity analysis," *Pattern Recognition*, vol. 90, pp. 346–362, 2019.
- [29] X.-N. Ye, K.-H. Liu, and S.-T. Liong, "A ternary bitwise calculator based genetic algorithm for improving error correcting output codes," *Information Sciences*, vol. 537, pp. 485–510, 2020.
- [30] Y.-P. Zhang, X.-N. Ye, K.-H. Liu, and J.-F. Yao, "A novel multi-objective genetic algorithm based error correcting output codes," *Swarm and Evolutionary Computation*, vol. 57, Article ID 100709, 2020.

- [31] T. G. Dietterich and G. Bakiri, "Solving multiclass learning problems via error-correcting output codes," *Journal of Artificial Intelligence Research*, vol. 2, pp. 263–286, 1994.
- [32] M. Galar, A. Fernández, E. Barrenechea, H. Bustince, and F. Herrera, "NMC: nearest matrix classification - a new combination model for pruning One-vs-One ensembles by transforming the aggregation problem," *Information Fusion*, vol. 36, pp. 26–51, 2017.
- [33] K. Crammer and Y. Singer, "On the learnability and design of output codes for multiclass problems," *Machine Learning*, vol. 47, no. 2-3, pp. 201–233, 2002.
- [34] L. Zhou, Q. Wang, and H. Fujita, "One versus one multi-class classification fusion using optimizing decision directed acyclic graph for predicting listing status of companies," *Information Fusion*, vol. 36, pp. 80–89, 2017.
- [35] F. Masulli and G. Valentini, "Effectiveness of error correcting output coding methods in ensemble and monolithic learning machines," *Formal Pattern Analysis & Applications*, vol. 6, no. 4, pp. 285–300, 2004.
- [36] D. Dutta, J. Sil, and P. Dutta, "Automatic clustering by multi-objective genetic algorithm with numeric and categorical features," *Expert Systems with Applications*, vol. 137, pp. 357–379, 2019.
- [37] M. A. Bagheri, Q. Gao, and S. Escalera, "A genetic-based subspace analysis method for improving Error-Correcting Output Coding," *Pattern Recognition*, vol. 46, no. 10, pp. 2830–2839, 2013.
- [38] L. I. Kuncheva and L. C. Jain, "Designing classifier fusion systems by genetic algorithms," *IEEE Transactions on Evolutionary Computation*, vol. 4, no. 4, pp. 327–336, 2000.