Online Teaching Design and Evaluation of Innovation and Entrepreneurship Courses in the Context of Education Internationalization

Chengshe Xing
Yiwu Industrial & Commercial College, School of Foreign Studies, Yiwu, 322000, China

***Abstract*—In the context of the internationalization of education nowadays, courses in innovation and entrepreneurship have been strongly promoted, and the content and number of topics, etc. of this type of courses are rapidly climbing. In order to enable target users to quickly select courses that they may be interested in, one changed collaborative filtering algorithm based on a multi-feature ranking model is used to extract and rank the features of online courses based on several factors, and then combine the collaborative filtering algorithm to recommend them to users. The results of experiment show that the numerical valuation of accuracy rate and recall rate of the improved algorithm are more than those of the other algorithm with different conditions, and in most cases higher than those of the LDA algorithm, and the user's evaluation of the recommendation effect also has the highest rating value of the improved algorithm, with the ratings of 4.3, 4.7 and 4.4 in the three groups, and the overall average score is 4.47, indicating that the improved algorithm has significant optimization performance and is suitable for teaching innovation and entrepreneurship in online courses.**

***Keywords*—Online course; Collaborative filtering algorithm; Ranking model**

I INTRODUCTION

Traditional teaching suffers from single teaching mode, unbalanced teaching resources and low teaching efficiency. As the process of education informatization accelerates, more and more information technology and intelligent algorithms are applied to the education field and improve the existing teaching quality from them. With the increase of people's reliance on online teaching, educational resources have started to grow explosively in recent years, and many domestic and foreign experts have started to research on personalization of educational resources. The application of recommendation models to online teaching can not only improve the learning efficiency of learners and save their time in finding materials, but also improve the existing teaching models and educational resources. Based on this, recommendation models have become a hot research topic in the field of education. In recent years, innovation and entrepreneurship courses have been of high interest and attention, and in the context of the current internationalization of education, the content required for innovation and entrepreneurship courses has been increasing, which has led to an increasing demand for online courses in related directions [1]. The industry of teaching online courses has become increasingly mature, and course recommendations are often made in online courses to shorten the time of user selection or to give suggestions for them [2]. For teaching recommendations in online courses, the core of the recommendation system is the recommendation algorithm used, and the performance of the recommendation algorithm directly affects the final recommendation results [3]. There are several recommendation methods, among which the most commonly used are different types of collaborative filtering algorithms. Collaborative filtering algorithms have the advantages of fast computational rate and high accuracy, but also have drawbacks such as cold start problem [4]. Therefore, in order to make the collaborative filtering algorithm fully applicable to the teaching recommendation of courses, the traditional collaborative filtering algorithm must be improved and achieve the desired effect. Currently, collaborative filtering is one of the most mature techniques in recommender systems, which uses the similarity of interests or features to find the nearest neighbors and then recommend the target users. Although this algorithm has high recommendation quality, it still faces the problem of data sparsity, where users generate more evaluations for mature domains and fewer evaluations for unfamiliar or new domains, thus causing the cold start phenomenon. If we can learn users' preferences or features from mature domains and use them in the recommendation of new domains, it will greatly alleviate the problem of data sparsity in new domains.

II RELATED WORK

For the research and improvement of collaborative filtering algorithm, many experiments have been successfully conducted by scholars at home and abroad. Liu came up to a changed clustering-based collaborative filtering algorithm for reference by introducing a function of decay time and item attribute vector and characterizing items and user interest vector to describe users, and projecting recommendation candidate sets in clusters, and the result of experiment showed that the algorithm can solve the data sparsity and problem of new items [5]. Xu et al. established object meta-classification by introducing a new dependency function based on Gaussian kernel and extended classification method for information of input statistics data, and recommended results by computing the dependency between the features of classification set and the features set of target objects. The results of experiment show that, compared with conventional algorithms, the new hybrid has higher speed and better performance [6]. Chen's team calculates the similarity between users by combining the collaborative filtering algorithm with other algorithms, and computes users' calligraphy words in addition to the main recommended calligraphy words based on the preliminary recommendation results to get the final recommendation results [7]. Panda team designed a collaborative filtering recommender based on normal filter for recommender systems to recommend personalized objects to strict users. The algorithm determines the average user rating for each object, computes the number of users who bought corresponding objects. Then uses min-max normalized way to find the number of users who have been normalized for each object in a specific range to scale the average user rating, and finally tested and found to predict user ratings more accurately [8]. Yu et al. came up to a cross-domain algorithm based on feature collaborative filtering construction and locally weighted linear regression by constructing features in different domains and using these features to represent different auxiliary domains. Also, they used a locally weighted linear regression model to solve the regression problem. Results of experiment show that this regression algorithm effectively solves the data sparsity problem by transferring useful data of knowledge from the auxiliary features domains [9].

Jiang's team proposed a slope-one algorithm to calculate the similarity between users by selecting trusted data and adding this similarity to the weighting factor of the changed algorithm to obtain the final recommendation equation. The base of new algorithm is the fusion of trusted data. User similarity under collaborative filtering algorithm acts more accurately than traditional algorithms [10]. Osval Montesinos-López et al. developed a project-based collaborative filtering package for multi-trait and multi-environment data and used it to study the prediction accuracy of precious data under phenotypic and genomic selection. The results of simulation experiment showed that package was more accurate for studying genomic prediction and data predictions are more accurately [11]. Zhang et al. designed a coverage-based collaborative filtering algorithm to provide brilliant recommendations for new users, improved previous collaborative filtering by reconstructing a decision class with detailed analysis of new user characteristics, and used a coverage-based the results showed that the improved algorithm significantly outperformed the existing working algorithm [12]. Li et al. in view of the increasingly serious problem of information overload and the fact that the traditional recommendation algorithm does not take the social relationship of users as the basis of recommendation, a combination algorithm with social information and dynamic time window is proposed. Through dynamic time window comparison, the time function is introduced to determine the corresponding time weight of user interest at different times. Finally, the practicability and effectiveness of the proposed method are verified. The experimental results show that the performance of the proposed algorithm is better than that of the traditional collaborative filter synthesis algorithm [13]. Yildirim studied the relevance between users through online social networks and used a multi-type improved collaborative filtering algorithm used for shopping recommendations, and the experimental results showed that the numerical valuation of accuracy rate under the changed algorithm recommendations is higher [14]. Han et al. based on the improved k-means clustering of small batches, an improved time weighted collaborative algorithm based on small batch K-means is proposed. The algorithm combines Pearson correlation coefficient with k-means algorithm, uses the improved k-means clustering algorithm of small batch to cluster the sparse scoring matrix, and introduces Newton cooling time weighting to improve user similarity. The experimental results are obviously superior to the traditional algorithm in all aspects [15].

From the above research results, it can be found that there are a large number of studies related to the personalization technology of collaborative filtering algorithm, and a considerable number of studies in different fields are used to improve the traditional collaborative filtering algorithm, but there are relatively few studies on the personalized teaching technology based on algorithm about collaborative filtering, so the research is based on the personalization technology of algorithm about collaborative filtering to design an innovative entrepreneurial network for each target user. Therefore, the research is based on the personalization technology of algorithm about collaborative filtering to design an innovative entrepreneurship web course for each target user to enhance the learning interest and learning ability of target users in the context of internationalization of education.

III Improved Collaborative Filtering Algorithms based on Multi-Feature Extraction Ranking Model and its Application

A. *Application of Collaborative Filtering Algorithm and Ranking Model to Online Courses*

Along with the rapid development of internationalization of education, the content of innovation and entrepreneurship courses that can be learned, gradually increased, and at the same time the personalized requirements of course learning also increased. In this context, experiments are conducted to study the personalized recommendation of courses, and the collaborative filtering recommendation algorithm is widely used in the recommendation algorithm and is suitable for the context of internationalization of education under big data. There are two tasks in the collaborative filtering recommendation system, which are rating prediction and Top-N recommendation. The rating prediction is mainly to predict the rating of items not rated by users according to their characteristics, while the Top-N recommendation recommends the N most likely items of interest to users based on the rating and the rating prediction.

Collaborative filtering algorithms are classified into memory-based collaborative filtering algorithms, model-based collaborative filtering algorithms, and content-based collaborative filtering algorithms. Due to the current explosive growth in the amount of information on the Web, the rapid increase in the information available to users makes a single type of collaborative filtering algorithm no longer applicable, and practical applications also usually mix multiple types of methods to adapt to more complex practical situations [16]. While in Top-N recommendation mainly neighbor models, the most commonly used one is the K-nearest neighbor model. K-nearest neighbors use the K most similar items or user ratings for weighting and use them as a basis to predict user ratings for unknown items [17]. The nearest neighbor algorithm focuses on the similarity calculation, and the closer the similarity result is 0 to 5, the higher the similarity is, and the closer the users or objects are to each other. The cosine similarity and Pearson coefficient for similarity calculation are shown in equation (1) and equation (2), respectively.

$$\omega\cos(y,z)=\frac{r_y\cdot r_z}{\left|r_y\right|^2\cdot\left|r_z\right|^2} \tag{1}$$

$$\omega pearson(y,z)=\frac{\sum b\in B(r_{y,b}-\overline{r}_y)(r_{z,b}-\overline{r}_z)}{\sqrt{\sum b\in B(r_{y,b}-\overline{r}_y)^2}\sqrt{\sum b\in B(r_{z,b}-\overline{r}_z)^2}} \tag{2}$$

In Eq. (1) and (2), r_y and r_z are the target vectors of the neighboring targets y and z, respectively. Cosine similarity is suitable in the absence of ratings by and loss vector space pinch cosine values for similarity judging, while Pearson coefficient is suitable in the case of both ratings and can eliminate rating noise and minimize the influence of users differing in rating stringency [18]. After completing the similarity calculation, the aggregation method is used to predict the numbers of rating on unknown objects as shown in equations (3) and (4).

$$r_{c,s}=\frac{1}{\sum_{c`\in U}\left|sim(c,c`)\right|}\sum_{c`\in U} sim(c,c`)\cdot r_{c`,s} \tag{3}$$

$$r_{c,s}=r_c+\frac{1}{\sum_{c`\in U}\left|sim(c,c`)\right|}\sum_{c`\in U} sim(c,c`)\cdot(r_{c`,s}-r_{c`}) \tag{4}$$

Both r_c and $r_{c`}$ in Eq. (3) and (4) are the average ratings of the target users $c$ and $c`$, respectively. Eq. (3) represents the aggregation function that does not consider the difference in users' rating styles, while Eq. (4) corrects the aggregation function for the difference in rating styles by calculating the deviation between the weighted and used rating values and the corresponding mean score difference values of the target users. The similarity determination recommendation process is shown as Fig. 1.

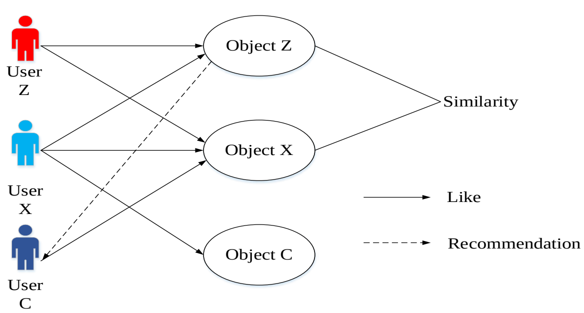

Fig. 1. User based collaborative filtering algorithm.

In practical applications, algorithms about collaborative filtering need to extract project features and target user features, and the extraction of user features is actually a supervised classification problem, so classification algorithms in machine learning can be used, including decision tree algorithms and linear classification algorithms, in addition to the nearest neighbor algorithm [19]. So, in order to be able to design a reasonable web-based distance course, the experiment requires feature extraction of course items and course learning users and reasonable recommendations using collaborative filtering algorithms, etc. The formula for the target user's number of rating of the item is shown in equation (5). In equation (5), the set of nearest neighbors is represented by S_u , $\overline{R}_u$ and $\overline{R}_n$ mean the average ratings of users u and n on the objects each, the similarity on two users is represented by $sim(u,n)$, and the ratings of users n on the item j are represented by $R_{n,j}$.

$$P_{u,j}=\overline{R}_u+\frac{\sum_{n\in S_u} sim(u,n)\cdot(R_{n,j}-\overline{R}_n)}{\sum_{n\in S_u} sim(u,n)} \tag{5}$$

The content-based collaborative filtering algorithm will make the k most similar items (x1, x2, x3, x4 ....) based on the similarity of the target item x evaluated by target user .xk) get their corresponding similarity S (i.e., Sx1, Sx2, Sx3, Sx4 ..... .Sxk), and then obtain a weighted average, which is based on the ratings of all those similar items by the target users, and then with the similarity of the target items as the weights, and use this result as the final rating of the target items by the target users. Finally, a number of top items are selected to be recommended to users based on the magnitude of the predicted value. A simple schematic of this algorithm is shown in Fig. 2.

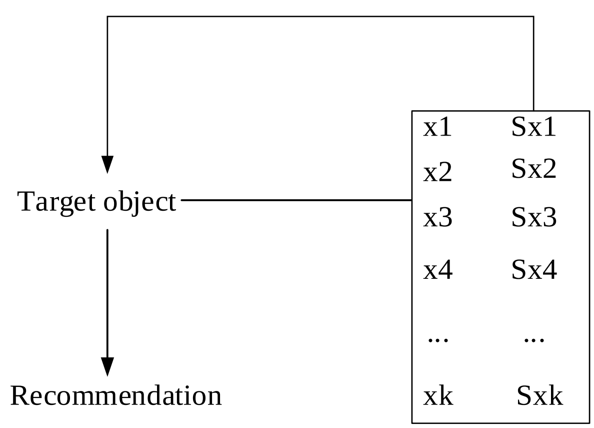

Fig. 2. Object based collaborative filtering algorithm.

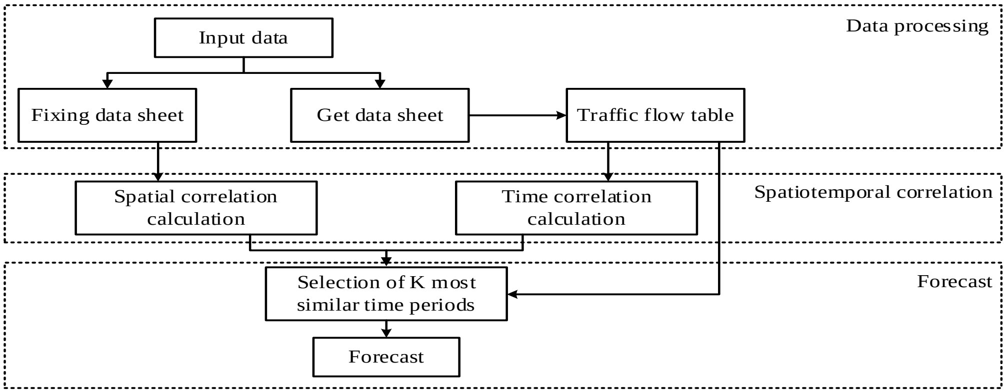

Fig. 3. Recommended schematic diagram of K proximity.

In order to fit the demands of the online teaching process for the main purpose, the dataset of the online course contains five data tables, and they are basic course information, course chapter information, basic user information, user learning and collection records and course creator information. Before the normal operation of the online course, the course information, course creator and user information need to be analyzed, at this time, the method of ranking learning can be used, and the ranking learning is integrated into the recommendation algorithm, and the weight parameters of multiple ranking models are learned by machine learning methods, and the best combination model is obtained in the training set to get better recommendation results [20]. Sorting learning usually requires the use of support vector machines, which are mainly divided into three types of methods: point-level methods, pair-level methods, and list-level methods, and their collaborative filtering k-neighborhood recommendation process is shown in Fig. 3.

The experiments are chosen to transform the ranking problem into a classification problem using a pairwise ranking method with fast training speed and moderate training complexity. Suppose from x1 to xn are the feature vectors of documents, which are from d1 to dn , at this point define a new training sample, where the positive samples are from x1-x2 to x1-xn, negative samples are from x2-x1 to xn-x1 .Then a binary classifier is trained to classify these new samples. The document classification uses the support vector machine approach with a linear scoring function as shown in equation (6).

$$f(x) = \omega^t (x_u - x_v) \tag{6}$$

In equation (6) ω is the marginal term and t is the time variable parameter. This method is integrated into the recommendation algorithm of course items, where the target users are ranked according to their preferences for the course items, and a matrix of users' access or learning time for the course items is established, and the input of the algorithm is a user-program evaluation matrix $R(m,n)$ as shown in equation (7). The rows and columns in Eq. (7) represent the user and the item, respectively, and the element in the R_{ij} , i row of the j column in the matrix refers to the user's i numbers of rating on the object j .

$$R(m,n) = \begin{bmatrix} R_{1,1} & R_{1,2} & R_{1,3} & R_{1,4} & \cdots & R_{1,5} \\ R_{2,1} & R_{2,2} & R_{2,3} & R_{2,4} & \cdots & R_{2,5} \\ \vdots & \vdots & \vdots & \vdots & \vdots & \vdots \\ R_{m,1} & R_{m,2} & R_{m,3} & R_{m,4} & \cdots & R_{m,5} \end{bmatrix} \tag{7}$$

B. Improvement of Collaborative Filtering Algorithm Based on Multi-Feature Ranking Model

Conventional collaborative filtering algorithms based on ranking learning still have their limitations due to the shortcomings of traditional collaborative filtering algorithms such as cold start problem, sparsity problem and data density problem. Therefore, we designed an improved collaborative filtering algorithm based on multi-feature ranking and tested its applicability to the instructional design of an online course [21]. Considering each type of collaborative filtering algorithm, the experiment uses a user-based nearest neighbor recommendation algorithm. The similarity between all users is calculated, not by mixing all but by combining any two users. The similarity between the two is calculated by defining the collaborative filtering preference value of user x for course i as shown in equation (8). In equation (8), $r(y,i)$ represents the rating of course 2 items by user y and i is the total number of users.

$$pref(x,i) = \frac{\sum_{y=1}^{k} sim(x,y) \cdot r(y,i)}{\sum_{y=1}^{k} sim(x,y)} \tag{8}$$

The obtained preference value results range from 1 to 5, and the higher the value is, means the higher the user's favorite preference for the course item. This preference value is used as a preference feature and applied in feature extraction, which is mainly associated with the calculation of similarity of the administrators, the time complexity of the calculation process and the number from users and items [22]. Since in the learning of online courses, courses with similar contents are not repeatedly studied by users, so in order to maximize the expression of users' interest bias, the experiment uses a subject model for content preference feature extraction, i.e., a subject-based collaborative filtering algorithm for user preferences. Firstly, the course text information is abstracted into a subject vector, with the name and introduction as the course information, and the course learning record, personal filled-in interests and user learning record as the user profile. The course information and user profile are combined. The distribution of the courses that users have studied on the subject vector space is calculated as a weighted average, after which the similarity between the feature vector about users and the feature vector of courses is calculated [23]. After performing the similarity calculation, the user feature vectors and the subject-based user preferences are calculated as shown in Eqs. (9) and (10). In equation (9) and equation (10), γ_h represents the feature vector about the course h , γ_g is the feature vector of the corresponding user, γ_h` and $\gamma_g$` are the same, and m is the number of courses that the corresponding user has taken.

$$\gamma_g = \frac{1}{m}\sum_{h=1}^{m}\gamma_h \tag{9}$$

$$pref\,(g,h) = \frac{\gamma_g \gamma_h`}{\sqrt{(\gamma_g \gamma_g`)(\gamma_h \gamma_h`)}} \tag{10}$$

In addition to the above-mentioned features, feature extraction also takes into account the influence of the course lecturer itself and the popularity of the course. Since the influence of a course instructor cannot be directly quantified in the short term, the combined popularity of all the courses of the instructor is used as the influence of the instructor, and the popularity of the course should be studied first. The extraction of course popularity features should take something account number which is the numerical results of learners with user ratings, the number of scores with the ratio of the number of scores to the number of learners of the course. The feature extraction for calculating the popularity of a course is shown in equation (11).

$$val(u,i) = 0.2c_i + s_i \cdot d_i \tag{11}$$

In equation (11), c_i denotes the numerical valuation of learners for the course i , d_i denotes the number of raters for the course i , and s_i denotes the all to aver numerical valuation after rating the course i . Since ratings are usually on a 5-point scale, to balance the any possible influence of the numerical results of learners and ratings on the course popularity value, the numerical result of learners is multiplied by a factor of 0.2. After deriving the absolute score of popularity, the calculation of similarity on popularity is shown in equation (12).

$$pref\,(u,i) = -\frac{r_i}{m} + 1 \tag{12}$$

In equation (12), m is the summary number of classes under the respective category, and r_i indicates the rank number on the course i in terms of the absolute value of its popularity among these courses. Similarly, the similarity value ranges from 0 to 1 and the closer to 1 means higher the similarity.

After obtaining the similarity of popularity, we can obtain the influence and popularity of the corresponding tutors as shown in equation (13). In equation (13), $prefteach(u,i)$ represents the similarity of influence of tutors, $prefhot(u,j)$ represents the similarity of popularity of different courses taught by tutors, m is the total number of courses, and C is a coefficient to consider the relationship between users and tutors, because users may have different values of C depending on whether they follow a tutor or not, and whether they have already taken a course. If you have not taken a lesson with a tutor and do not follow them, the value is 1.

$$prefteach(u,i) = C \cdot \frac{1}{m}\sum_{j=1}^{m} prefhot(u,j) \tag{13}$$

After all features are extracted and similarity is calculated, these recommendations are integrated using multi-feature ranking learning. A simple schematic representation of machine integration learning is shown in Fig. 4.

For a given user and project course, each pair of relationship is reflected as a vector x , each dimension in the vector which in this situation refers to the different features extracted, and the vector dimension is the number of features, at this time the ranking function is $f(x) = Wx$ and W is the weight vector [24]. In this model, training sets are established according to the user learning of different courses, and the courses are grouped in each two items, and the first i group in the training set is shown in equation (14).

$$D_i = (xi_1 - xi_2), i \in N^* \tag{14}$$

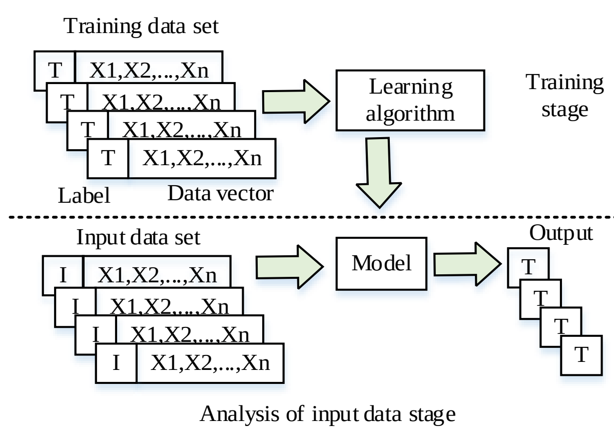

Fig. 4. Simple schematic diagram of machine sequencing calculation.

For each two sets of courses, the difference of the feature vectors of the two course items is used as the sample marker, and the samples that cannot be sorted by the sorting relationship are ignored, and they are divided into two sets according to the different features of the two courses, which are positive and negative samples, after the sorting training is performed on these sample data [25]. The loss function for the ranking training is shown in equation (15). The same in equation (15) ω is the marginal term and t is the time variable parameter.

$$lossf = \min \sum_{i=1}^{m} \max[1-\omega^t(xi_1 - xi_2)] + \frac{1}{2}\|\omega\|^2 \quad (15)$$

The learning process of machine group sorting is then shown in Fig. 5. After that, the desired result of the predecessor is calculated.

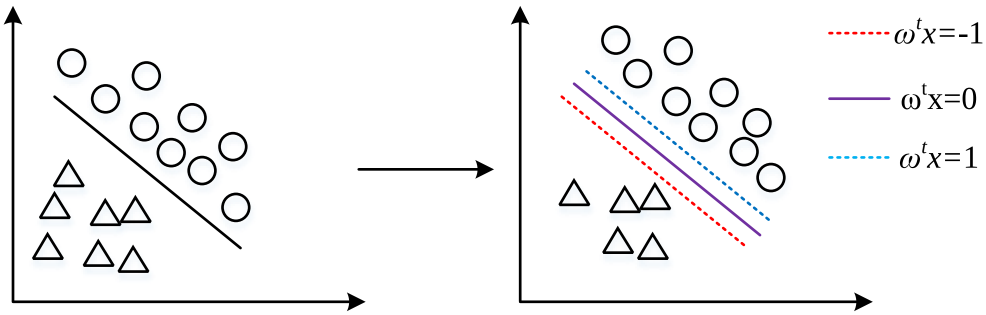

Fig. 5. Simple diagram of support vector machine learning.

IV DISCUSSION

The above research method optimizes the traditional collaborative filtering model and finally designs an improved collaborative filtering algorithm based on multi-feature ranking. In order to test the performance of the algorithm and explore whether the improved collaborative filtering algorithm is suitable for online teaching course recommendation, this section of the research content firstly selects a series of evaluation indexes for evaluating the strengths and weaknesses of different algorithms, and then compares the performance of evaluation indexes of different algorithms under the same test data set. The commonly used evaluation metrics for prediction scoring accuracy include mean absolute error, root mean square error, and standardized mean absolute error.

$$MAE = \frac{1}{I_T} \sum_{(u,i)\in I_T} \left| r_{ui} - \tilde{r}_{ui} \right| \quad (16)$$

Eq. (16) shows the formula of Mean Absolute Error (MAE) MAE, where I_T represents the test set. r_{ui} represents the actual rating of item i by user u. $\tilde{r}_{ui}$ represents the predicted rating of item i by user u. The Mean Absolute Error can show the absolute error between the predicted and actual ratings, and the smaller the value, the better the recommendation effect of the algorithm.

$$RMSE = \sqrt{\frac{1}{I_T} \sum_{(u,i)\in I_T} \left(r_{ui} - \tilde{r}_{ui} \right)^2} \quad (17)$$

Eq. (17) represents the calculation formula of root mean square error (RMSE) RMSE.

$$NMAE = \frac{MAE}{r_{\max} - r_{\min}} \quad (18)$$

Eq. (18) is the calculation formula of normalized mean absolute error (NMAE) NMAE. $r_{\min}$ indicates the maximum value of the scoring range. $r_{\min}$ indicates the minimum value of the scoring range. According to the above three average indicators, the commonly used recommendation algorithm and the improved algorithm in this paper are tested.

TABLE I. PERFORMANCE OF EVALUATION INDICATORS OF DIFFERENT ALGORITHMS IN THE SAME DATA SET

Recommended algorithm	MAE	NMAE	RMSE
Collaborative Filtering	26.54	13.27	5.15
Apriori	29.28	14.64	5.41
K-Means	18.35	9.18	4.28
BPNN	24.65	12.33	4.96
Algorithm in text	12.56	6.28	3.54

Table I shows five different types of recommendation algorithm models, including traditional collaborative filtering algorithm, Apriori algorithm based on association rules, K-Means algorithm based on clustering recommendation, Back Propagation Neural Network (BPNN) BPNN and improved collaborative filtering algorithm proposed in the paper. The MAE, NMAE and RMSE values of the five algorithms under the same recommendation data set are compared. According to Table I, the MAE values of the five algorithm models are 26.54, 29.28, 18.35, 24.65 and 12.56 respectively; NMAE values are 13.27, 14.64, 9.18, 12.33 and 6.28 respectively; The RMSE values are 5.15, 5.41, 4.28, 4.96 and 3.54 respectively. It can be seen that the improved collaborative filtering algorithm proposed by the research has good recommendation performance. Next, the algorithm is applied to the recommendation of online teaching resources for innovation and entrepreneurship courses, and the recommendation effect of collaborative filtering algorithm before and after improvement is compared. The accuracy and recall rate are used to evaluate the recommendation accuracy of the recommendation model.

TABLE II. RELATIONSHIP BETWEEN RECOMMENDATION RESULTS AND USERS

User information	System recommendation	The system does not recommend
Users are interested	True-Positive（N_{tp}）	False-Negative（N_{fn}）
User is not interested	False-Positive（N_{fp}）	True-Negative（N_{tn}）

Table II shows the user's response to the recommendation system. As shown in Table II, there are four situations. That is, the user is interested in the recommended content of the recommendation system. The user is not interested in the recommended content of the recommendation system. The system does not recommend to the user but the user is interested. The system does not recommend to the user and is not interested.

$$P = \frac{N_{tp}}{N_{tp} + N_{fp}} \tag{19}$$

Eq. (19) shows the calculation formula of the recommended accuracy of the model. Accuracy can clearly represent the recommended performance of the model.

$$R = \frac{N_{tp}}{N_{tp} + N_{fn}} \tag{20}$$

Eq. (20) shows the calculation formula of the recommended recall rate of the model. The recall rate can indicate the probability that the content that users are interested in is recommended.

V EXPERIMENTAL RESULTS AND COMPARATIVE ANALYSIS OF THE THREE ALGORITHMS

A. Experimental Results of Feature Extraction and Model Training Prior

The experiment calculates user preferences based on common interval and topic-based user preferences, and randomly tries 80% of the training set and 20% of the training set. User course matrix test set. In order to calculate user preferences based on public filtering, use the marked value of KNN positive parameter K to capture the curve between the same value K, as shown in Fig. 6. Fig. 6 shows that the attack curve K is out of position. In order to fully study the needing rate, count and pick the value K 20 for the experiment.

Using Linear Discriminant Analysis (LDA) to calculate topic-based user preferences, the topic is the most important parameter in the algorithm, the experiment will be LDA as a separate recommendation algorithm for each user to select the corresponding topic vector as the most similar course as a result of the recommendation for that user, the number of topics and LDA recall results obtained are shown in Fig. 7. It can be seen that the recall rate gets one small increase with increase of the number of topics, and the number of topics with a number of 100 is used for comprehensive consideration in Fig. 7.

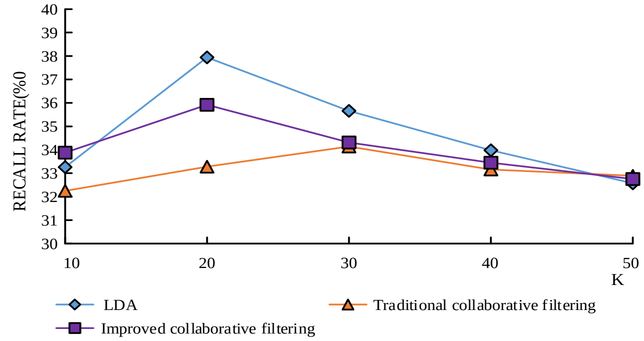

Fig. 6. Parameter K curve about recall rate.

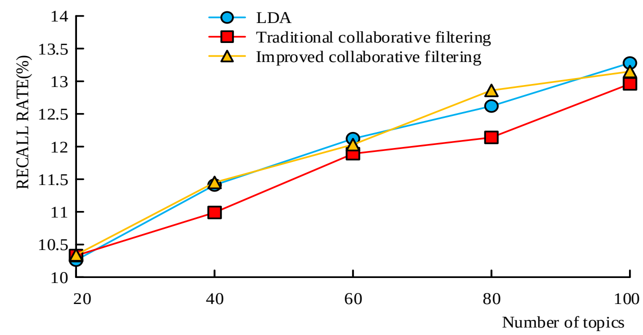

Fig. 7. Result curve of LDA about the number of topics.

B. Analysis and Evaluation of the Results for the Online Course Algorithm

After learning the ranking function at the training, a ranking model is used to recommend courses according to their ranking and consequently generate the user’s interest labels. The algorithm is then tested experimentally, along with two other algorithms, the regular algorithm about collaborative filtering and the linear discriminant analysis algorithm, and the effectiveness of the algorithms is measured and the results compared mainly by accuracy and recall.

The accuracy results of the three algorithms with different number of recommendations are displayed in Fig. 8. It can be seen, which are the truth that the accuracy rates of the three algorithms show a significant upward trend with the increase in the number of recommendations, which means that the recommendation algorithms should increase the number of recommendations from Fig. 8. When the number of recommendations is small, the accuracy rate of the improved algorithm is significantly higher than that of the traditional algorithm and the LDA algorithm, and when the number of recommendations gradually increases, the accuracy rate of the improved algorithm is still significantly higher than that of the traditional collaborative filtering, and at this time, although the accuracy rate is higher than that of the LDA algorithm, it will no longer be significant as the number of recommendations increases, indicating that algorithm which came up from the research is significantly better than the traditional algorithm and better than the LDA in general.

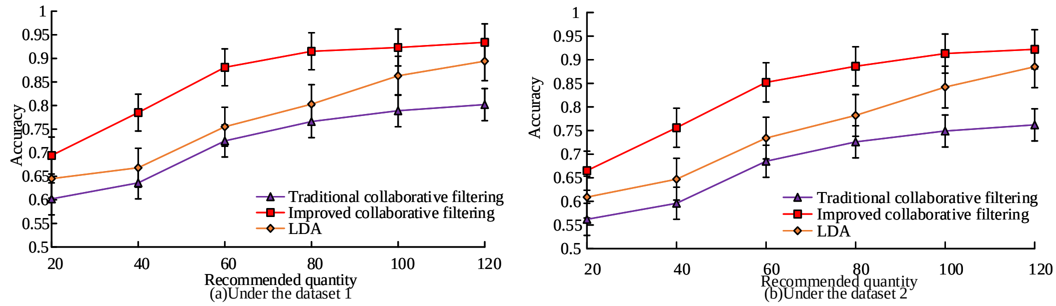

Fig. 8. Recommended quantity and accuracy results of three algorithms.

Fig. 9 shows the results of canceling many of the three proposed algorithms. As shown in Fig. 9, the cancellation rate of these three algorithms also increases with the increase of the number of recommendations. Among the three algorithms, the improved filter synthesis algorithm always has higher inverse speed than the traditional algorithm. When the recommendation number is low, the inverse value of LDA Algorithm is lower than the improved algorithm. The difference between LDA output coefficient and standard algorithm output coefficient gradually decreases, and the amount of recommendation increases, slightly higher than the improvement rate of the algorithm; among them, 120 recommendations.

For the same number of comments, the impact of the number of consumer training on the repeatability results is shown in Fig.10. Fig. 10 shows that the download rate of the three algorithms usually increases with the increase of the number of user training. When the number of user training is more, the calculation can play a better role. The extraction rate of the improved link filtering algorithm between the three algorithms is much higher than that of the traditional algorithm, and is always higher than that of the LDA Algorithm. When the number of training users increase, the download rate of the improved algorithm is significantly higher than the LD rate. The table improved algorithm is better than the other two algorithms, and when the number of courses increases in a certain amount of pasta, the advantage of the improved algorithm is greater.

The impact of the number of user learning courses on the accuracy results is shown in Fig. 11. From Fig. 11, it can be seen that the accuracy of the three algorithms shows an overall upward trend as the number of user-learning courses increases, indicating that the algorithms can play a better recommendation effect when the number of user-learning courses is more. The accuracy of the improved algorithm among the three algorithms is always higher than the other two algorithms, and the difference between the improved algorithm and the LDA is more significant as the number of courses increases, indicating that the improved algorithm will have a superior performance as the number of courses increases.

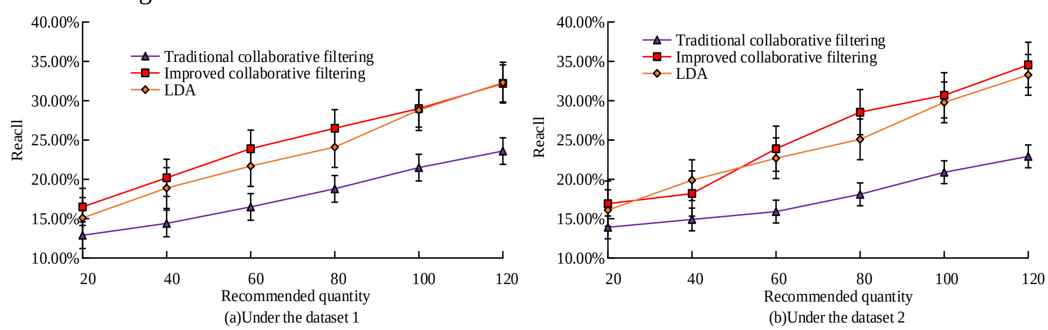

Fig. 9. Results of recommended quantity and recall rate of three algorithms.

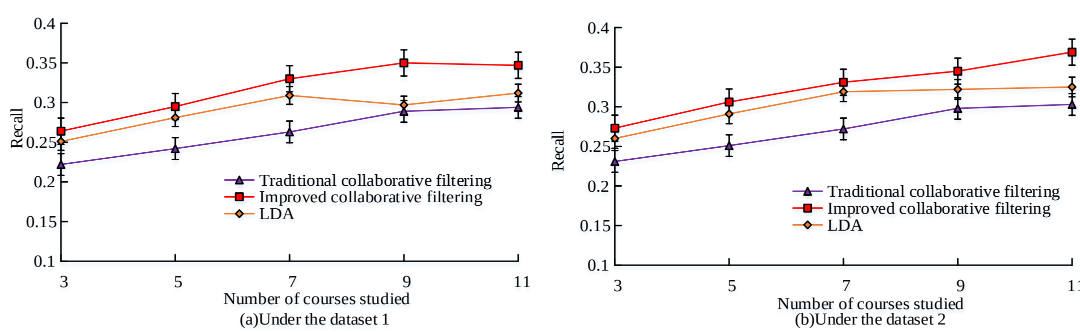

Fig. 10. Results of number of learning courses and recall rate of three algorithms.

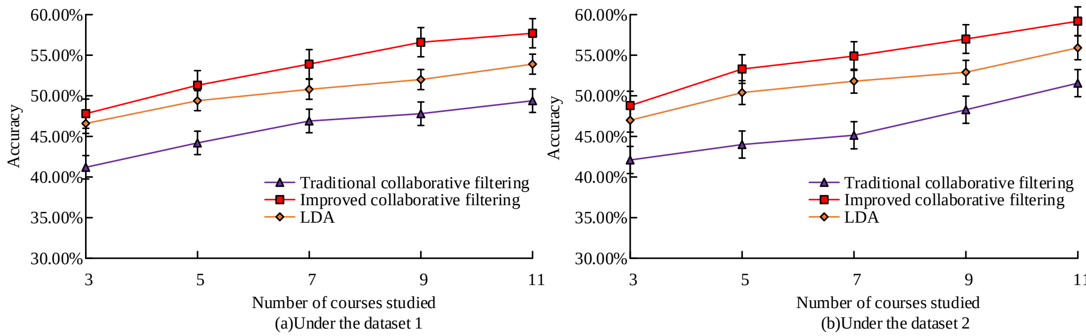

Fig. 11. Results of number of learning courses and accuracy of three algorithms.

The results of the recommendations were transmitted to the target users, the target users were divided into two large flat-rate groups, the results of the evaluation of the target users were taken from the sample and the results of each algorithm assessment were broken down into an average of five groups and then the average result for each group was calculated, - obtain the results of the evaluation referred to in Fig. 12. Fig. 12 shows that, with the exception of the third group, the improved algorithm rates are higher than the other two and that the difference between the improved algorithm and the third group algorithm is lower, which fully indicates that the improved algorithm designs a course with higher performance than the class and is suitable for online courses training.

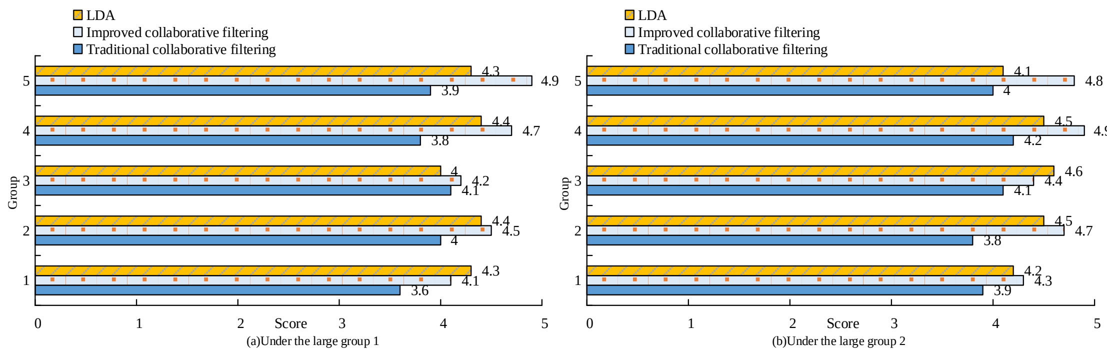

Fig. 12. Comparison of evaluation results of three algorithms.

VI CONCLUSION

The research uses a modified cooperation filtering algorithm based on a multifunctional classification model, which is tested and compared with the regular cooperation filtering algorithm and LDA algorithm after sample training. Experimental results show that the maximum accuracy and recall rate of the improved algorithm is 92.2% and 32.2% respectively for different recommended quantities which are significantly higher than the traditional algorithm; the maximum level of accuracy with an improved algorithm cancellation rate is 57.7% and 34.7% respectively for different numbers of learning courses which are significantly higher than the other two algorithms. The overall mean of the improved algorithm is 4.46, which is higher than the other two algorithms in 5 with 5. Experimental results show that the improved algorithm works much better than the traditional algorithm and, in many cases, better or much better than the LDA algorithm. Although some results have been obtained in the study, the total number of samples selected in the experiments is still small and the overall shortage of random sampling results due to the small number of samples is the main direction for further studies and optimization in the future.

VII FUTURE WORK

In this study, an improved collaborative filtering algorithm based on a multi-feature ranking model is used to construct a recommendation model for teaching innovation and entrepreneurship online courses in the context of internationalization of education. After testing the performance of the model, it is found that the constructed model has better recommendation effect, higher accuracy rate, and higher user satisfaction for innovation and entrepreneurship online courses. Although the research conducted in this paper has achieved certain results and optimized the traditional collaborative filtering algorithm to a certain extent, the following shortcomings still exist.

1) Although the improved collaborative filtering algorithm is applied to the course recommendation model, it may cause some bias to the experimental results because more pre-processing is not performed on the adopted educational data set.

2) Whether the constructed recommendation algorithm model is applicable to other e-learning recommendations for different majors is not explained in the paper. Subsequent research should attempt to apply the model to more online course recommendations for different majors.

3) In the current recommendation field, there are also more studies on the optimization of collaborative filtering algorithm, and the subsequent research should combine more advanced algorithms to optimize collaborative filtering and thus improve the recommendation accuracy.

ACKNOWLEDGEMENT

The research is supported by: 2022 Key research project of Yiwu Industrial & Commercial College: Teaching Evaluation Research in Higher Institutions against Education Internationalization (ZD2022WY322-01).

REFERENCES

[1] N. Lin, Y. Lin, "Research on the Integration of Innovation and Entrepreneurship and Ideological and Political Courses in Universities under the Background of Internet Era", Journal of Physics: Conference Series, 2021, vol. 1852(4), pp. 42033-42037, 2021.

[2] J. Lee, L. Martin, "Investigating Students' Perceptions of Motivating Factors of Online Class Discussions", International Review of Research in Open & Distance Learning, 2017, vol. 18(5), pp. 1492-1496, 2017.

[3] W. Zhang, X. Zhou, W. Yuan, "Collaborative Filtering Algorithm Based on Improved Time Function and User Similarity", Journal of Physics: Conference Series, vol. 1757(1), pp. 12080-12087, 2021.

[4] D. F. Meng, N. Liu, M. X. Li, et al. "An Improved Dynamic Collaborative Filtering Algorithm Based on LDA", IEEE Access, vol. 99(2), pp. 32-37, 2021.

[5] X. Liu, "An improved clustering-based collaborative filtering recommendation algorithm" Cluster Computing, vol. 20(2), pp. 1281-1288, 2017.

[6] L. B. Xu, X. S. Li, Y. "Guo Gauss-core extension dependent prediction algorithm for collaborative filtering recommendation", Cluster Computing, vol. 22(4), pp. 11501-11511, 2019.

[7] S. Chen, L. Huang, Z. Lei, et al. "Research on personalized recommendation hybrid algorithm for interactive experience equipment", Computational Intelligence, vol. 36(3), pp. 1348-1373, 2020.

[8] S. K. Panda, S. K. Bhoi, M. Singh, "A collaborative filtering recommendation algorithm based on normalization approach", Journal of Ambient Intelligence and Humanized Computing, vol. 11(1), pp. 4643-4665, 2020.

[9] X. Yu, Q. Peng, L. Xu, et al. "A selective ensemble learning based two-sided cross-domain collaborative filtering algorithm" Information Processing & Management, 2021, vol. 58(6), pp. 102691-102702, 2021.

[10] L. Jiang, Y. Cheng, L. Yang, et al. "A trust-based collaborative filtering algorithm for E-commerce recommendation system", Journal of ambient intelligence and humanized computing, vol. 10(8), pp. 3023-3034, 2019.

[11] A. Osval, "Montesinos-López, Francisco Javier Luna-Vázquez, Abelardo Montesinos-López, et al. An R Package for Multitrait and Multienvironment Data with the Item-Based Collaborative Filtering Algorithm", The Plant Genome, vol. 11(3), pp. 1-16, 2018.

[12] Z. Zhang, Y. Kudo, T. Murai, et al. "Improved covering-based collaborative filtering for new users' personalized recommendations", Knowledge and Information Systems, vol. 62(5), pp. 3133-3154, 2020.

[13] D. Li, C. Wang, L. Li, et al. "Collaborative filtering algorithm with social information and dynamic time windows", vol. 52(3), pp. 261-272, 2021.

[14] S. Z. M. Yildirim, "A new similarity coefficient for a collaborative filtering algorithm", Communications Faculty of Science University of Ankara, vol. 59(2), pp. 41-54, 2017.

[15] X. Han, Z. Wang, H. J. Xu, "Time-Weighted Collaborative Filtering Algorithm Based on Improved Mini Batch K-Means Clustering", Advances in Science and Technology, vol. 1059(1), pp. 309-317, 2021.

[16] N. Liu, M. X. Li, H. Y. Qiu, et al. "A hybrid user-based collaborative filtering algorithm with topic model", Applied Intelligence, vol. 12(51), pp. 7946-7959, 2021.

[17] C. Ajaegbu, "An optimized item-based collaborative filtering algorithm", Journal of Ambient Intelligence and Humanized Computing, vol. 51(1), pp. 5261-5272, 2021.

[18] H. Wang, Z. Shen, S. Jiang, et al. "User-based Collaborative Filtering Algorithm Design and Implementation", Journal of Physics: Conference Series, vol. 1757(1), pp. 012168-012173, 2021.

[19] Y. Fan, H. Ma, Z. Chen, et al. "Research and Application of Algorithm Based on Maximum Expectation and Collaborative Filtering in Recommended System", Journal of Physics Conference Series, vol. 1754(1), pp. 012205-012209, 2021.

[20] A. A. Alwan, H. Ibrahim, N. I. Udzir, "A Model for Ranking and Selecting Integrity Tests in a Distributed Database", International Journal of Information Technology & Web Engineering, vol. 5(3), pp. 65-84, 2017.

[21] Y. Yang, H. Yao, R. Li, et al. "A collaborative filtering recommendation algorithm based on user clustering with preference types", Journal of Physics: Conference Series, vol. 1848(1), pp. 012043-012049, 2021.

[22] Wang L. Application Model of Collaborative Filtering Algorithm Recommended for Pollution Source Information [J]. Software Engineering and Applications, 2020, 09(5):345-351.

[23] R. Jenke, A. Peer, M. Buss, "Feature Extraction and Selection for Emotion Recognition from EEG", IEEE Transactions on Affective Computing, vol. 5(3), pp. 327-339, 2017.

[24] K. L. Mak, K. Yiu, Z. Feng, "Feature extraction of the patterned textile with deformations via optimal control theory", Discrete and Continuous Dynamical Systems - Series B (DCDS-B), vol. 16(4), pp. 1055-1069, 2017.

[25] M. E. Banihabib, M. H. "Shabestari Fuzzy Hybrid MCDM Model for Ranking the Agricultural Water Demand Management Strategies in Arid Areas", Water Resources Management, vol. 31(1), pp. 495-513, 2017.