

## Research Article

# Personalized Course Recommendation System Fusing with Knowledge Graph and Collaborative Filtering

Gongwen Xu <sup>1</sup>, Guangyu Jia,<sup>1</sup> Lin Shi,<sup>1</sup> and Zhijun Zhang<sup>2</sup>

<sup>1</sup>School of Business, Shandong Jianzhu University, Jinan 250101, China

<sup>2</sup>School of Computer Science and Technology, Shandong Jianzhu University, Jinan 250101, China

Correspondence should be addressed to Gongwen Xu; xugongwen@sdjzu.edu.cn

Received 31 August 2021; Accepted 16 September 2021; Published 26 September 2021

Academic Editor: Ahmed Mostafa Khalil

Copyright © 2021 Gongwen Xu 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.

Personalized courses recommendation technology is one of the hotspots in online education field. A good recommendation algorithm can stimulate learners' enthusiasm and give full play to different learners' learning personality. At present, the popular collaborative filtering algorithm ignores the semantic relationship between recommendation items, resulting in unsatisfactory recommendation results. In this paper, an algorithm combining knowledge graph and collaborative filtering is proposed. Firstly, the knowledge graph representation learning method is used to embed the semantic information of the items into a low-dimensional semantic space; then, the semantic similarity between the recommended items is calculated, and then, this item semantic information is fused into the collaborative filtering recommendation algorithm. This algorithm increases the performance of recommendation at the semantic level. The results show that the proposed algorithm can effectively recommend courses for learners and has higher values on precision, recall, and F1 than the traditional recommendation algorithm.

## 1. Introduction

Recently, with the integration of Internet and education, various online education platforms have emerged and developed rapidly. These platforms have accumulated many users with their high-quality and massive resources. Especially affected by the COVID-19, the online education all over the world has developed rapidly. The daily active users of Coursera, edX, XuetangX, and other online education platforms have reached more than 10 million. The explosive growth of online education has changed the traditional teaching mode and made it a reality for people to learn anytime and anywhere. Online education has become an important educational model and technical way for learners to acquire knowledge and expand skills and academic education [1].

However, nowadays a wide range of learning directions and all kinds of courses have sprung up on the online education platform. The massive courses also bring confusion to learners and have some disadvantages for learners. Firstly, it is difficult for learners to find suitable courses in a short

time; secondly, the learning route that learners are interested in is complex, while the courses in the school training program tend to be single and repetitive, so the online learners' satisfaction is low and cannot achieve the expected teaching effect; thirdly, most online education platforms do not provide the same effective learning guidance and curriculum planning for learners as the traditional education. Learners are usually short of an in-depth understanding of the overall knowledge structure, while the number of Internet learning resources is miscellaneous, so learners fall into a large number of course choices, resulting in information overload and even the low course passing rate [2, 3].

How to provide personalized content for learners in massive courses resources is a problem worthy of research. Mining the interests of learners on the online education platform would help to better understand the needs of learners and help the platform provide personalized teaching services for learners. As the solution to this problem, the traditional interest mining methods either take the courses that learners are interested in as the interest point, or take the topic of the discussion area as the interest point. So many

online education teams at home and abroad are also discussing and studying this problem [4, 5]. There are many research results obtained in recent years, but the recommendation courses are often based on the history of students' choice of courses and the big data analysis results of popular courses. Although the recommendation results are also practical, they lack the support for the whole learning process of learners and lack the recommendation in combination with the training scheme of the school.

Personalized recommendation technology has been widely used in various fields, such as news headlines, online videos, "we media" short video, and so on, which effectively solves the problem of information overload in various fields [6]. As the recommendation algorithm of knowledge graph as side information has been fully concerned and studied, it is possible for the courses recommendation algorithm based on knowledge graph to further improve the recommendation effect [7].

## 2. Overview of the Course Recommendation Algorithm

*2.1. The Course Recommendation Algorithm.* Course personalized recommendation technology is one of the research hotspots in the field of online education and big data education. Many research teams all over the world have put forward the concept of personalized online education and studied personalized course recommendation algorithms in order to reduce the dropout rate of online learning, stimulate and mobilize learners' enthusiasm for active learning, and give full play to the learning personality of different learners.

Bhaskaran and Santhi mined learners' browsing logs according to learners' behaviour and preferences and used the hybrid recommendation strategy proposed by ApriorAll algorithm to realize personalized recommendation for learners [8]. Based on the traditional data mining methods, Obeidat et al. [9] and others used collaborative filtering method and comparison of association rules to recommend courses for learners. The experimental results showed that grouping and clustering learners can significantly improve the recommendation effect. Huang et al. used reinforcement learning method and Markov decision process [10] to recommend exercises for learners. When recommending, they comprehensively considered the smoothness of exercise difficulty, review and preview, and the degree of learners' participation. Liu et al. proposed using the method based on neural network [11] to track learners' knowledge level, so as to provide learners with personalized learning path recommendation. The MCRs proposed by Zhang et al. are based on the distributed association rule mining algorithm [12], which makes the recommendation information transmitted more timely and improves the user's course retrieval efficiency.

Chen et al. carried out a knowledge recommendation algorithm based on learners' existing knowledge and learning materials [13], which modelled the

recommendation process as a Markov decision problem. Aguilar et al. analysed the differences and similarities between the course selection system and the e-commerce platform, improved knowledge, and discovered association rules through biological heuristic algorithms, while most of the research in recent years were to explore the behaviour characteristics of learners to represent the characteristics of learners, so as to produce recommendation results [14].

Pang et al. extracted the behaviour characteristics of each learner and converted them into vectors of the same dimension, and dispersed them among similar users, so these learners would have more same courses [15]. They proposed multilayer bucket recommendation and MapReduce technology extension, so as to increase the satisfaction of the traditional collaborative recommendation algorithm.

To sum up, there have been a lot of works in learner personalized course recommendation recently years.

*2.2. The Problems in the Traditional Course Recommendation Algorithm.* At present, the existing research works mainly focused on using collaborative recommendation or data mining methods to improve the accuracy of course recommendation, but there are some problems in these methods, such as the cold start of recommendation algorithm, building a recommendation model without good performance, and carrying out an algorithm which cannot accurately recommend without initial data [16]. These problems are described as follows:

- (1) Accuracy of course recommendation: the recommendation algorithm using highly accurate rate can provide learners with more suitable courses and enhance their satisfaction. The accuracy of recommendation results is an important index of recommendation algorithm. If the recommendation result generated by a recommendation algorithm does not bring out good recommendation results, the recommendation results will be meaningless in some certain [17].
- (2) The problem of data sparsity: when using the traditional collaborative filtering method, the similarity calculation mainly depends on the learners' rating matrix of the course. In the actual process, the number of the course rating score is very small; in some cases, the scoring users account for only 2% of the total users. It makes the course scoring matrix so sparse, and it is difficult for courses and learners to find similar neighbors, resulting in the low quality of courses recommendation [18].
- (3) Cold start problem: now the course platform would constantly update the content under the needs of new learners and new courses, but there is no record of new learners or new course content before updating the course platform, resulting in the failure of the recommendation algorithm to make timely and effective recommendations [19].

### 3. Related Research

**3.1. Recommendation Based on Collaborative Filtering.** Collaborative filtering algorithm is widely used in recommendation field. Traditional recommendation algorithms ignore the characteristic data of users or items in advance and only rely on the historical behaviour data of users to build models and recommend items to users. In most collaborative filtering algorithms, nearest neighbor technology is used to calculate the distance between users with the help of historical preference information [20]. The algorithm uses the weighted score of the nearest neighbor of the target user to predict the target user's preference for specific courses, but it often faces problems such as sparse data and mining interpretation of recommendation results.

Collaborative filtering has been successfully applied in different researching fields [21]. The collaborative filtering recommendation system is based on the idea that similar users may have similar preferences or similar items may be preferred by similar users [22]. Collaborative filtering is generally divided into two categories:

- (1) User-based algorithm [23]: according to the users' behaviour record, it would find the user set which is similar in the interest and preference of the target user, then find the items liked by each user in this user set, filter out the items that the target user has already selected, and finally recommend the remaining items to the target user. In this method, the user similarity matrix calculated by the similarity formula is used to predict the target user's score on the items liked by similar users.
- (2) Item-based algorithm [24]: according to the users' behaviour records, it calculates the similarity between items and filters out the item set with high score among the item sets generated by the target user. Then, according to the item similarity matrix, the other items, most similar to each item in the collection, would be found out, sorting and filtering the items that the target user has selected at the same time, and then finally the rest items are recommended to the target user.

The collaborative filtering algorithm uses the weighted score of the nearest neighbor of the target user to predict the target user's preference for specific courses, but sometimes it would face the problems of sparse data and unexplained recommendation results.

**3.2. Recommendation Method Based on Knowledge Graph.** Google put forward the concept of knowledge graph, which uses the "entity-relation-entity" 3-tuple to describe the semantic relationship between different entities in the real world and forms a network knowledge structure through the relationship [25]. Representation learning aims to transform the objects into the low-dimensional space [26], and the aim of knowledge graph representation learning is to map the entities and relationships in the knowledge graph. The

resulting vector can effectively represent the semantic relationship between entities and relationships [27].

Recently, the representation learning methods of deep learning have attracted extensive attention [28–34]. The common models of knowledge graph representation learning include distance model, energy model, matrix decomposition model, bilinear model, translation model, and so on [35]. The translation model represented by Ebisu and Ichise [36] has fewer parameters and is more popular in the research fields. For each 3-tuple  $(h, r, t)$ , where  $h$  and  $t$  represent head entities and tail entities, respectively,  $r$  is the relationship between them, TransE represents  $h$ ,  $t$ , and  $r$  as the embedded vectors  $v_h$ ,  $v_t$ , and  $v_r$ , respectively, and  $v_r$  is the translation from vectors  $v_h$  to  $v_t$ , and the relationship among the three elements is expressed as follows:

$$v_h + v_r \approx v_t. \quad (1)$$

In the TransE model, it aims to train vectors  $v_h$ ,  $v_t$ , and  $v_r$  effectively to make formula (1) infinitely close to be equal. The smaller the error between them, the more likely there is a relationship  $r$  between the head and tail. Therefore, the loss function is shown as follows:

$$f(v_h, v_r, v_t) = \|v_h + v_r - v_t\|_2^2, \quad (2)$$

where  $\|\cdot\|_2$  is the 2-norm of the vector, that is, the Euclidean distance. The total cost function for all 3-tuple samples is expressed as follows:

$$L = \sum_{(h,r,t) \in S} \sum_{(h',r,t') \in S'} \max(0, f(v_h, v_r, v_t) - f(v_{h'}, v_r, v_{t'}) + \delta), \quad (3)$$

where  $S$  is positive sample and is the set of all 3-tuple in the knowledge graph;  $S'$  is the negative samples of  $S$ , that is, each existing 3-tuple in  $S$  is randomly replaced with its head entity or tail entity to obtain a new 3-tuple, and the 3-tuple does not belong to  $S$ ; then the set of such 3-tuples is called negative sample; and  $\delta$  is the distance between  $S$  and  $S'$ . In order to minimize the cost function  $L$  during training process, it is necessary to make the loss function  $f(v_h, v_r, v_t)$  of positive samples tend to 0 and the loss function  $f(v_{h'}, v_r, v_{t'})$  of negative samples tend to infinity.

### 4. Recommendation Algorithm Fusing with Knowledge Graph and Collaborative Filtering Technology

**4.1. The Algorithm Model.** At present, collaborative filtering recommendation algorithms usually use users' historical evaluation data without considering the semantic relationship between recommendation items. To solve this problem, based on previous studies, a courses resource recommendation algorithm fusing knowledge graph and collaborative filtering technology (FKGCF) was proposed. This algorithm directly integrates the semantic similarity of the recommended object into the similarity calculation of the collaborative filtering recommendation algorithm and makes up for the deficiency that this algorithm ignores the connotation characteristics of

the item itself from the semantic perspective. The algorithm model is shown in Figure 1.

**4.2. Composition of Course Knowledge Graph.** The interaction information between learners and courses in the existing data sets is sparse or even missing, which will lead to the cold start problem. This problem can be alleviated by introducing other information, namely, side information, into the recommendation algorithm. Side information introduced in common recommendations such as movies and commodities includes social networks, user/item attributes, multimedia, and contexts [4]. The side information used in this paper is the knowledge graph, which is represented by 3-tuple  $(C_1, R, C_2)$ . The knowledge graph of the course is  $G$ :

$$G = \{(c_1, r, c_2) | c_1, c_2 \in C, r \in R\}, \quad (4)$$

where  $c_1, c_2 \in C$ ,  $C$  is the set of all the courses, and  $r \in R$ ; there are 5 relations in  $R$ : same.instructor, same.subject, hour.low, hour.mid, and hour.high. When a learner chooses some certain courses, these courses can be connected to other events in the knowledge graph, connected to many other nonitem entities, and then connected to other course entities from these nonitem entities. For example, if there are computer courses in the selected courses, it can be related to other computer courses belonging to the same subject. Then, a 3-tuple knowledge graph is built.

In the traditional collaborative filtering algorithm, the connection is established by other learners' historical learning records and other interactive processes. The essence of the knowledge graph constructed by the FKGCF algorithm is to establish a connection between the courses which learners have interacted and the courses which they have not interacted. These connections are not obtained from the interaction history of other learners, but through non-item entities. The method of constructing knowledge graph provides additional information sources connected between courses and a more accurate calculation method of item similarity in the algorithm, so as to improve the accuracy of recommended courses [37].

**4.3. The Semantic Similarity.** The entities and relationships in the knowledge graph are embedded into a  $d$ -Dimensional semantic space by using the TransE algorithm, and the item semantic vector is expressed as follows:

$$I_i = (E_{1i}, E_{2i}, \dots, E_{di})^T, \quad (5)$$

where  $I_i$  presents the semantic vector of item  $I$  and  $E_{ki}$  is the value of the  $k$ th dimension semantic vector, where  $1 \ll k \ll d$ . The TransE algorithm trains the loss function based on Euclidean distance. The similarity of item semantics is also measured by this distance, and the calculation equation is as follows:

$$d(I_i, I_j) = \sqrt{\sum_{k=1}^d (E_{ki} - E_{kj})^2}. \quad (6)$$

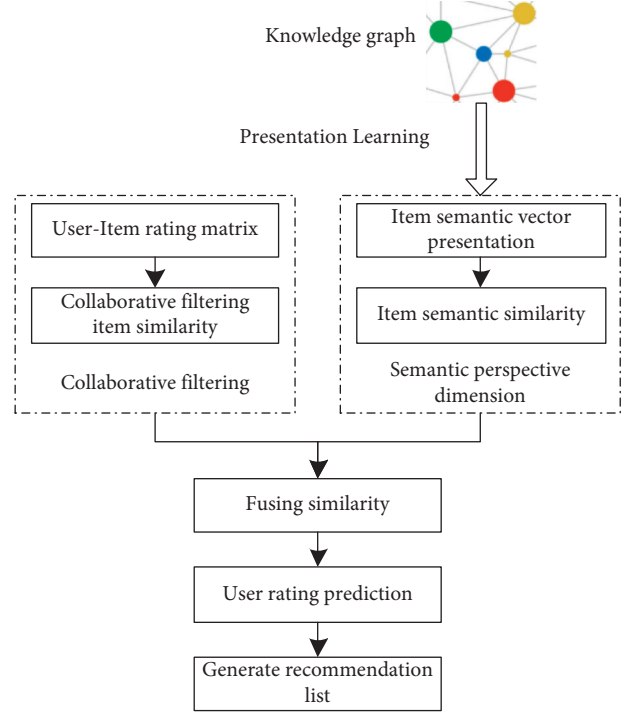


FIGURE 1: FKGCF algorithm model.

In order to unify the value of distance within the range of  $(0, 1)$ , the following calculation method is carried out:

$$\text{sim}_{\text{sem}}(i, j) = \frac{1}{1 + d(I_i, I_j)}. \quad (7)$$

The greater the value of  $\text{sim}_{\text{sem}}(i, j)$  is, the more similar the semantics of items  $i$  and  $j$  is.

In the knowledge graph, these pieces of feature information form a “entity-relationship-entity” 3-tuple. The more similar the semantic information is, the closer the course vector is, and the more similar the users' love for it. Using these rich semantic data, the cold start problem of recommendation system is solved to a certain extent.

**4.4. Fusing Similarity.** After the item semantic similarity  $\text{sim}_{\text{sem}}$  was obtained by knowledge graph representation algorithm, the similarity in the collaborative filtering recommendation algorithm was calculated, as shown in

$$\text{sim}_{\text{CF}}(i, j) = \cos(S_i, S_j) = \frac{S_i \cdot S_j}{\|S_i\| \cdot \|S_j\|} = \frac{\sum_{k=1}^m r_{ki} \cdot r_{kj}}{\sqrt{\sum_{k=1}^m r_{ki}^2} \cdot \sqrt{\sum_{k=1}^m r_{kj}^2}}, \quad (8)$$

where  $\text{sim}_{\text{CF}}(i, j)$  is the cosine similarity of collaborative filtering of items  $i$  and  $j$ , and  $S_i$  and  $S_j$  are the user's scoring vectors for items  $i$  and  $j$ , respectively.

Next, we fuse two similarities,  $\text{sim}_{\text{CF}}(i, j)$  and  $\text{sim}_{\text{sem}}(i, j)$ , as shown in

$$\text{sim}(i, j) = \alpha \cdot \text{sim}_{\text{sem}}(i, j) + (1 - \alpha) \cdot \text{sim}_{\text{CF}}(i, j), \quad (9)$$



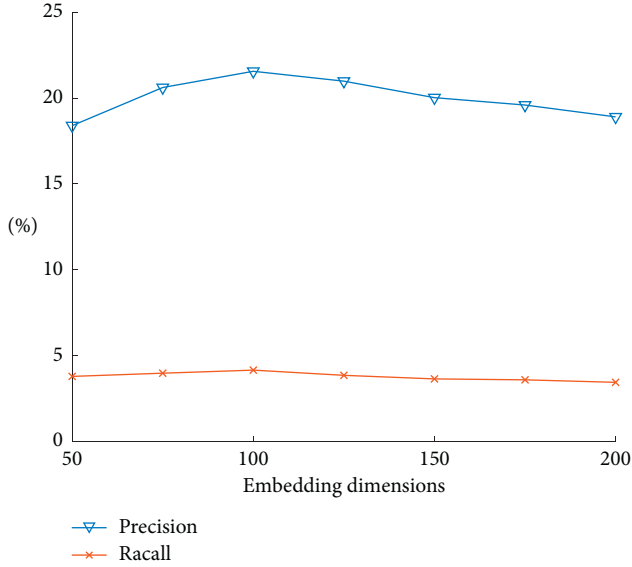


FIGURE 2: Results' comparison under different embedding dimensions.

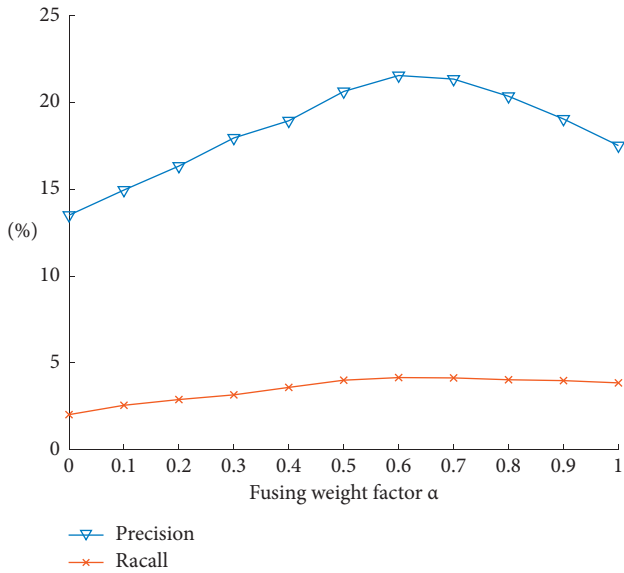


FIGURE 3: Results' comparison under different  $\alpha$  values.

where  $\text{sim}(i, j)$  represents fusing similarity of item  $i$  and  $j$ ,  $\text{sim}_{\text{sem}}(i, j)$  is the items semantic similarity,  $\text{sim}_{\text{CF}}(i, j)$  is the collaborative filtering item similarity, and  $\alpha$  is the weighted factor, whose value range is  $[0, 1]$ . When  $\alpha$  is 0, the algorithm is based on items. When  $\alpha$  is 1, the algorithm is based on semantic. When  $\alpha \in (0, 1)$ , it is the fusing recommendation algorithm. The collaborative filtering recommendation algorithm is to calculate the similarity of items indirectly from the perspective of users' rating of items.

**4.5. Rating Prediction and Generation of Recommendation List.** According to the fusing similarity, the user's score on an item that has not been evaluated before is predicted. The calculation equation of prediction score is as follows:

$$p_{ui} = \frac{\sum_{j \in (N(u) \cap S(i, k))} (\text{sim}(i, j) * r_{uj})}{\sum_{j \in (N(u) \cap S(i, k))} \text{sim}(i, j)}, \quad (10)$$

where  $p_{ui}$  represents the prediction score that user  $u$  gives to item  $i$ .  $N(u)$  refers to all item sets evaluated by user  $u$ ,  $S(i, k)$  refers to  $k$  adjacent items with the greatest similarity to item  $i$ ,  $N(u) \cap S(i, k)$  refers to the intersection of two item sets, and the intersection result is the reference for prediction score;  $\text{sim}(i, j)$  is the fusing similarity of items  $i$  and  $j$ ;  $r_{uj}$  is the user  $u$ 's rating of item  $j$ .

The higher the prediction score is, the more interested the user is. The prediction score of all items is calculated by using equation (10). And then they are sorted according to the scoring results, giving priority to the top- $N$  items recommended to the user.

## 5. Experimental Process

### 5.1. Experimental Environment and Data Set Processing.

The experimental data comes from XuetaangX. 1000 courses and 10000 members are collected for experiments. The precision, recall, and F1 value are used as evaluation indexes to measure the performance of each algorithm. The precision refers to the probability that the user is interested in the recommended course list, and recall refers to the probability that the course that the user is interested in appears in the recommendation list. F1 value is the harmonic average of precision and recall [38]. The calculating methods are shown as follows:

$$\begin{aligned} \text{Precision} &= \frac{TP}{TP + FP}, \\ \text{Recall} &= \frac{TP}{TP + FN}, \\ F1 &= \frac{2 \times \text{Precision} \times \text{Recall}}{\text{Precision} + \text{Recall}}. \end{aligned} \quad (11)$$

In order to measure the performance of the algorithm more accurately, this paper uses  $k$ -cross verification and the  $k$  value is 5; that is, the experimental data are randomly grouped in 5 parts, one of which is selected as the testing set and the other 4 as the training set. A total of 5 tests are carried out, and the average result of 5 tests is used as the final evaluation of this algorithm.

### 5.2. Experimental Process and Result Analysis

#### 5.2.1. Experiment of Different Embedding Dimensions.

When knowledge graph representation learning is carried on, different embedding dimensions will have different

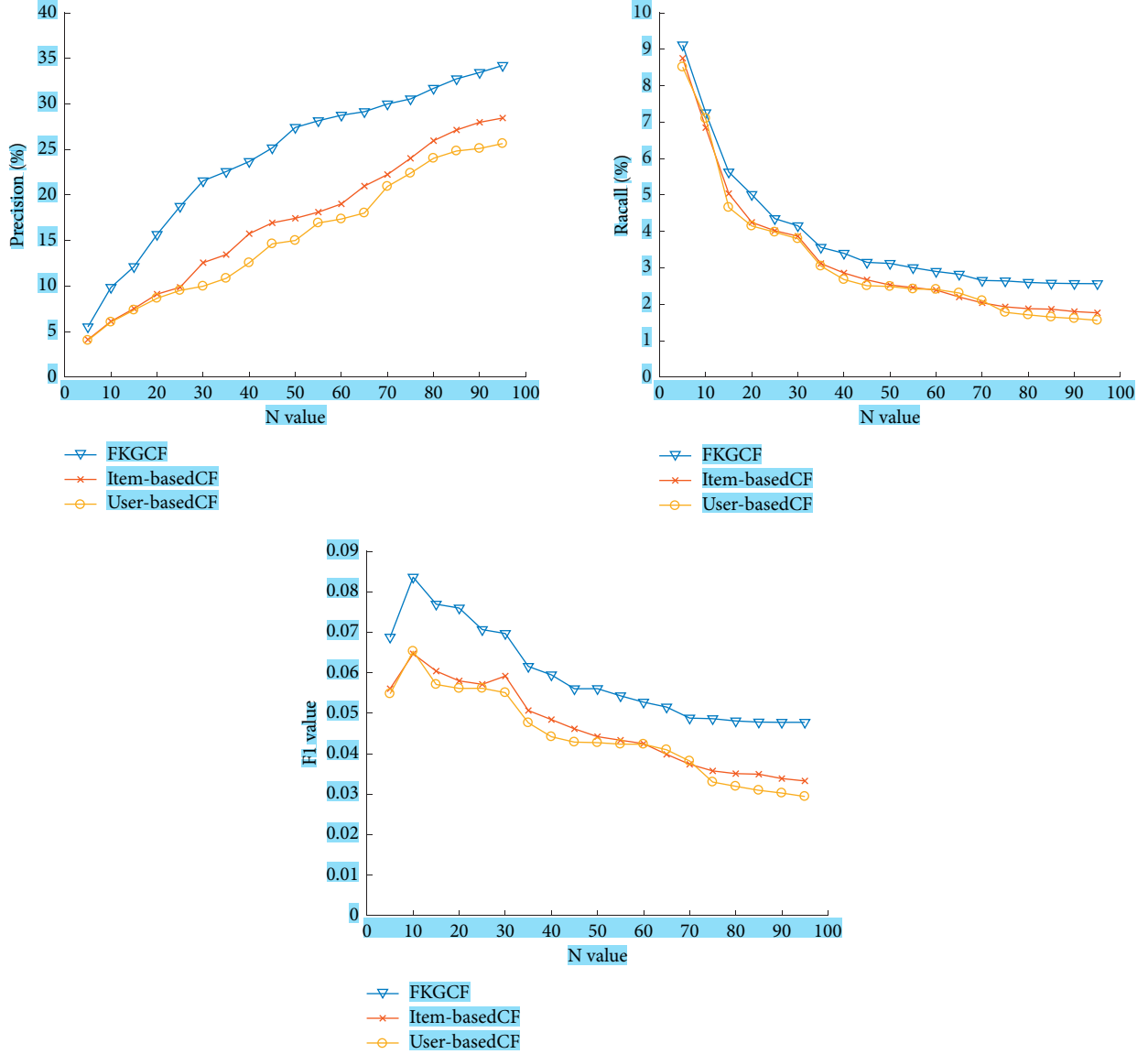


FIGURE 4: Experimental results (precision, recall, and F1 value).

results in the experiment. Seven dimensions, 50, 75, 100, 125, 150, 175, and 200, are selected for the experimental comparison. The similarity fusing weight factor  $\alpha$  is set to 0.6, and the maximum number of neighbors  $k$  of similarity is set to 30. The final experimental comparison results are shown in Figure 2.

From Figure 2, it can be found that there is no significant difference among the seven different embedding dimensions. But the precision and recall value are better when the dimension is 100.

**5.2.2. Experiment of Different Fusing Weight Factor Ratio.** Fusing weight factor  $\alpha$  controls the proportion of semantic similarity and collaborative filtering similarity in the final fusing similarity, and it is a key factor of this algorithm. The value range of  $\alpha$  is  $[0, 1]$ , and in the experiment  $\alpha$  is assigned from 0 to 1 with a step of 0.1, so there are 11 steps altogether.

The embedding dimension of knowledge graph presentation learning is set to 100, and the maximum number of neighbors  $k$  of similarity is set to 30. The results can be seen in Figure 3.

Experiments show that when the fusion reaches a certain proportion, the precision and recall of the proposed algorithm are higher than the single semantic content-based recommendation algorithm ( $\alpha=1$ ) and the item-based algorithm ( $\alpha=0$ ). When the fusion weight  $\alpha$  is 0.6, the precision and recall reach the best value.

**5.2.3. Comparison of Experimental Results.** To verify the performance of the modified collaborative filtering recommendation algorithm proposed in this paper, experimental comparisons are made with traditional algorithms, user-basedCF and item-basedCF. The embedding dimension of knowledge graph representation learning in this algorithm is

set to 100. The fusing weight  $\alpha$  is set to 0.6 and the recommended number top- $N$  is set to 10. The experimental results are shown in Figure 4. The  $N$  value refers to the number of neighbors in the nearest neighbor algorithm KNN. The figure illustrates that the precision of all comparative experiments increases with the increase of  $n$  value while the recall decreases, and the F1 value increases first and then decreases.

Through experimental comparison, it is obvious that the proposed FKGCF algorithm has higher precision, recall, and F1 value than the traditional algorithms.

## 6. Conclusions

Aiming at the problem that the collaborative filtering recommendation algorithm only uses the users' historical evaluation data without considering the semantic relationship between the recommended objects, a modified FKGCF algorithm was proposed. The algorithm not only uses the users' evaluation information of the course, but also uses the internal semantic information of the item itself. The results show that this algorithm improves the precision and recall and can realize the courses recommendation more accurately and efficiently. However, the algorithm in this paper also has some limitations, which does not consider the problem of user interest drift, because the users' interest is likely to change with the development of time, and the previous historical data is time-effective. Therefore, according to the time dynamics of historical data, establishing relevant time series recommendation models is a direction worthy of further research.

## 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 there are no conflicts of interest regarding the publication of this paper.

## Acknowledgments

This work was supported in part by the Shandong Education Department Teaching Reform Project (Z2016M014, Z2016M016, and Z2016Z013).

## References

- [1] China Internet Network Information Center (CNNIC), *The 46th Statistical Report on the Current Situation of Internet Development in China*, China Internet Network Information Center (CNNIC), Beijing, China, 2020.
- [2] J. Xiao, M. Wang, B. Jiang, and J. Li, "A personalized recommendation system with combinational algorithm for online learning," *Journal of Ambient Intelligence and Humanized Computing*, vol. 9, no. 3, pp. 667–677, 2018.
- [3] H. Zhang, H. Yang, T. Huang, and G. Zhan, "DBNCF: personalized courses recommendation system based on DBN in MOOC environment," in *Proceedings of the 2017 International Symposium on Educational Technology (ISET)*, pp. 106–108, Hong Kong, China, June 2017.
- [4] C. Yang, L. Bai, C. Zhang, Q. Yuan, and J. Han, "Bridging collaborative filtering and semi-supervised learning: a neural approach for poi recommendation," in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1245–1254, Halifax, Canada, August 2017.
- [5] J. Wen, X. Wei, T. He, and S. Zhang, "Regression analysis on the influencing factors of the acceptance of online education platform among college students," *Ingénierie des Systèmes d'Information*, vol. 25, no. 5, pp. 595–600, 2020.
- [6] Q. Zhang, Y. Liu, L. Liu, S. Lu, Y. Feng, and X. Yu, "Location identification and personalized recommendation of tourist attractions based on image processing," *Traitement du Signal*, vol. 38, no. 1, pp. 197–205, 2021.
- [7] Y. Xian, Z. Fu, S. Muthukrishnan, G. De Melo, and Y. Zhang, "Reinforcement knowledge graph reasoning for explainable recommendation," in *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*, pp. 285–294, Paris, France, July 2019.
- [8] S. Bhaskaran and B. Santhi, "An efficient personalized trust based hybrid recommendation (tbhr) strategy for e-learning system in cloud computing," *Cluster Computing*, vol. 22, no. 1, pp. 1137–1149, 2019.
- [9] R. Obeidat, R. Duwairi, and A. Al-Aiad, "A collaborative recommendation system for online courses recommendations," in *Proceedings of the 2019 International Conference on Deep Learning and Machine Learning in Emerging Applications (Deep-ML)*, pp. 49–54, Istanbul, Turkey, August 2019.
- [10] Z. Huang, Q. Liu, C. Zhai et al., "Exploring multi-objective exercise recommendations in online education systems," in *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pp. 1261–1270, Beijing, China, November 2019.
- [11] Q. Liu, S. Tong, C. Liu et al., "Exploiting cognitive structure for adaptive learning," in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*, pp. 627–635, Anchorage, AK, USA, August 2019.
- [12] H. Zhang, T. Huang, Z. Lv, S. Liu, and Z. Zhou, "MCRS: a course recommendation system for MOOCs," *Multimedia Tools and Applications*, vol. 77, no. 6, pp. 7051–7069, 2018.
- [13] Y. Chen, X. Li, J. Liu, and Z. Ying, "Recommendation system for adaptive learning," *Applied Psychological Measurement*, vol. 42, no. 1, pp. 24–41, 2018.
- [14] J. Aguilar, P. Valdiviezo-Díaz, and G. Riofrio, "A general framework for intelligent recommender systems," *Applied Computing and Informatics*, vol. 13, no. 2, pp. 147–160, 2017.
- [15] Y. Pang, Y. Jin, Y. Zhang, and T. Zhu, "Collaborative filtering recommendation for MOOC application," *Computer Applications in Engineering Education*, vol. 25, no. 1, pp. 120–128, 2017.
- [16] Y. Liu, H. Yang, G. Sun, and S. Bin, "Collaborative filtering recommendation algorithm based on multi-relationship social network," *Ingénierie des Systèmes d'Information*, vol. 25, no. 3, pp. 359–364, 2020.
- [17] J. Lin, H. Pu, Y. Li, and J. Lian, "Intelligent recommendation system for course selection in smart education," *Procedia Computer Science*, vol. 129, pp. 449–453, 2018.
- [18] Y. Tian, B. Zheng, Y. Wang, Y. Zhang, and Q. Wu, "College library personalized recommendation system based on hybrid recommendation algorithm," *Procedia CIRP*, vol. 83, pp. 490–494, 2019.

- [19] J. Gope and S. Jain, "A survey on solving cold start problem in recommender systems," in *Proceedings of the 2017 International Conference on Computing, Communication and Automation (ICCCA)*, pp. 133–138, Greater Noida, India, 2017.
- [20] J. Shen, T. Zhou, and L. Chen, "Collaborative filtering-based recommendation system for big data," *International Journal of Computational Science and Engineering*, vol. 21, no. 2, pp. 219–225, 2020.
- [21] V. Subramaniaswamy, R. Logesh, M. Chandrashekhar, A. Challa, and V. Vijayakumar, "A personalised movie recommendation system based on collaborative filtering," *International Journal of High Performance Computing and Networking*, vol. 10, no. 1-2, pp. 54–63, 2017.
- [22] A. Sattar, M. Ghazanfar, and M. Iqbal, "Building accurate and practical recommender system algorithms using machine learning classifier and collaborative filtering," *Arabian Journal for Science & Engineering*, vol. 42, no. 8, 2017.
- [23] Z. Tan and L. He, "An efficient similarity measure for user-based collaborative filtering recommender systems inspired by the physical resonance principle," *IEEE Access*, vol. 5, pp. 27211–27228, 2017.
- [24] P. Thakkar, K. Varma, V. Ukani, S. Mankad, and S. Tanwar, "Combining user-based and item-based collaborative filtering using machine learning," in *Information and Communication Technology for Intelligent Systems*, pp. 173–180, Springer, Singapore, 2019.
- [25] B. Jia, X. Huang, and S. Jiao, "Application of semantic similarity calculation based on knowledge graph for personalized study recommendation service," *Educational Sciences: Theory & Practice*, vol. 18, no. 6, 2018.
- [26] T. Ebisu and R. Ichise, "Toruse: knowledge graph embedding on a lie group," in *Proceedings of the Thirty-Second AAAI Conference on Artificial Intelligence*, New Orleans, LA, USA, November 2018.
- [27] M. Fan, Q. Zhou, T. F. Zheng, and R. Grishman, "Distributed representation learning for knowledge graphs with entity descriptions," *Pattern Recognition Letters*, vol. 93, pp. 31–37, 2017.
- [28] K. Xu, C. Li, Y. Tian, T. Sonobe, K. I. Kawarabayashi, and S. Jegelka, "Representation learning on graphs with jumping knowledge networks," in *Proceedings of the International Conference on Machine Learning*, pp. 5453–5462, Hanoi, Vietnam, June 2018.
- [29] G. Xu, X. Li, and Z. Zhang, "Semantic consistency cross-modal retrieval with semi-supervised graph regularization," *IEEE Access*, vol. 8, pp. 14278–14288, 2020.
- [30] W. L. Meng, C. Z. Mao, J. Zhang, J. Wen, and D. H. Wu, "A fast recognition algorithm of online social network images based on deep learning," *Traitement du Signal*, vol. 36, no. 6, pp. 575–580, 2020.
- [31] A. Çınar and M. Yildirim, "Classification of malaria cell images with deep learning architectures," *Ingénierie des Systèmes d'Information*, vol. 25, no. 1, pp. 35–39, 2020.
- [32] B. P. Doppala, D. Bhattacharyya, and M. Chakkravarthy, "Stratification of cardiovascular diseases using deep learning," *Revue d'Intelligence Artificielle*, vol. 34, no. 4, pp. 377–385, 2020.
- [33] G. Xu, X. Li, L. Shi, Z. Zhang, and A. Zhai, "Combination subspace graph learning for cross-modal retrieval," *Alexandria Engineering Journal*, vol. 59, no. 3, pp. 1333–1343, 2020.
- [34] S. Demircan and H. K. Örnek, "Comparison of the effects of mel coefficients and spectrogram images via deep learning in emotion classification," *Traitement du Signal*, vol. 37, no. 1, pp. 51–57, 2020.
- [35] Y. Lin, X. Han, R. Xie, Z. Liu, and M. Sun, "Knowledge representation learning: a quantitative review," 2018, <https://arxiv.org/abs/1812.10901>.
- [36] T. Ebisu and R. Ichise, "Generalized translation-based embedding of knowledge graph," *IEEE Transactions on Knowledge and Data Engineering*, vol. 32, no. 5, pp. 941–951, 2019.
- [37] H. Zarzour, Z. Al-Sharif, M. Al-Ayyoub, and Y. Jararweh, "A new collaborative filtering recommendation algorithm based on dimensionality reduction and clustering techniques," in *Proceedings of the 2018 9th International Conference on Information and Communication Systems (ICICS)*, pp. 102–106, Irbid, Jordan, April 2018.
- [38] G. Xu, A. Zhai, J. Wang, Z. Zhang, and X. Li, "Cross-media semantic matching based on sparse representation," *Technical Gazette*, vol. 26, no. 6, pp. 1707–1713, 2019.