Personalized Learning Path Generation based on Network Embedding and Learning Effects

Wan Li
Department of Computer Science
Beijing University of Posts and Telecommunications
Beijing, China
liwan_mail@163.com

Lei Zhang
Department of Computer Science
Beijing University of Posts and Telecommunications
Beijing, China
zlei@bupt.edu.cn

Abstract—**With the development of network technology and data mining technology, personalized learning path recommendation in intelligent teaching system has become an important issue. Choosing the right learning object and recommending it to learners is a challenge for online learning systems. At present, the problem faced by online learning is that the information overload causes the learner cannot find a learning resource suitable for himself, and the recommended learning resource cannot satisfy the individual needs of the learner. The recommendation system is an important means of information filtering and a very promising method for solving information overload problems. Therefore, adaptive learning path recommendation is considered to be an effective means to solve the above problems in online learning. In this paper, we propose a learning path generation algorithm based on network embedding and learning effects. First, we use the network embedding to learn the user's expression in order to judge the learner's similarity. Then we generate an adaptive learning path for current learners based on the similarity between all historical learners and current learners and the learning effects of historical learners. The experimental results show that the average grade of the learners using our recommended learning path has been greatly improved. (Abstract)**

Keywords- network embedding; personalized learning path; adaptive learn; learner model (key words)

I. INTRODUCTION

T The purpose of the personalized learning environment is to guide students' learning activities according to their learning characteristics. One of the main problems in personalized and adaptive learning is the learning path recommendation. In a personalized and adaptive learning environment, the system dynamically provides guidance for students' learning paths based on their learning characteristics and current learning status, so that students can achieve better learning outcomes. But now not all learning systems can meet the needs of users, because these systems use the traditional method, that is, assume that students have the same goals, abilities and prior knowledge, ignore the differences between learners, and provide learners with the same learning resources and learning paths.

SACS[1], AACS[2], PWIS[3], and Ahmad's neural network based online learning system[4] are better adaptive learning systems. In recent years, some new recommendation algorithms have been applied in the field of learning path recommendation, such as environment-aware recommendation[5] and Bayesian network reasoning[6].

The online learning system can collect information about all aspects of the learner, and evaluate the learner's learning effect. This is very helpful for the learning path recommendation. We can recommend suitable learning paths for current learners based on the learning characteristics and learning effects of other learners. After generating the learner representation, we can calculate the similarity between the learners. Then, based on the learning effect of the historical learners and the similarity between the learners, we recommend a personalized learning path for the current learners to improve the learner's learning effect.

In this paper, we first obtain the learner's feature vector through network embedding. The nodes in the network include learners and courses, in which learner has attributes such as academic qualifications, age, learner type, and learning purpose, and courses has attributes such as its own subject.

The remainder of this paper is organized as follows. Section II will introduce the related work of network embedding, user model and adaptive learning path recommendation. Section III will introduce our model structure and the specific details of method. Section IV discusses dataset, experiments and results. Finally, this paper ends with a conclusion and an outlook in Section V.

II. REALTED WORK

A. User Model

The user model can represent the characteristics of the user and is a combination of multiple attributes of the user. It includes the academic qualifications of the learner, age, prior knowledge, learning purpose, learner type, etc. The learner's attributes are dynamically changed, the user model for each learner is different from other users.

In general, the basic information of the user characteristics used to model the user includes name, age, gender, cognitive ability, academic qualifications, learning objectives, and activities during the learning process[7, 8]. Some specific characteristics of the user are also used for user modeling, such as learning style[9], learning characteristics, learning habits,

978-1-7281-0945-9/19/$31.00©2019 IEEE

learner personality[10], etc. The user models with specific characteristics of user are highly adaptable and personalized.

B. Network Embedding

Network Embedding (NE), also known as Graph Embedding: uses low-dimensional, dense, real-valued vectors to represent nodes in the network, so that the semantic relationships between nodes can be calculated without having to manually extract features (with adaptability), and can project heterogeneous information into the same low-dimensional space facilitates downstream calculations. The central idea of NE is to find a mapping function that converts each node in the network into a low-dimensional potential representation.

DeepWalk[11] is the first network embedding method proposed to use representation learning (or deep learning). DeepWalk bridges the gap between network embedding and word embedding by treating nodes as words and generating short random walk sequences as sentences. A neural language model such as Skip-gram can then be applied to these random walk sequences. The advantages of DeepWalk can be summarized as follows: First, it can generate random walks on demand, since the Skip-gram model is also optimized for each sample, the combination of random walk and Skip-gram makes DeepWalk an online algorithm. Secondly, DeepWalk applies deep learning to the data of the graph structure. The process of DeepWalk is shown in Figure 1.

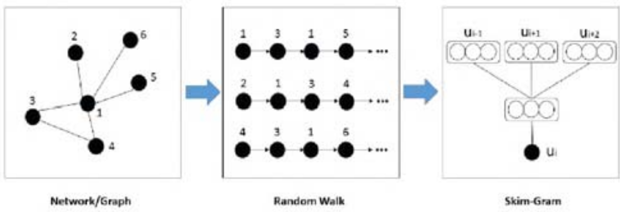

Figure 1. IThe schematic of DeepWalk model

LINE[12] uses a breadth-first search strategy to generate context nodes: only nodes that are up to two hops from a given node are considered to be neighbors. In addition, it uses negative sampling to optimize the Skip-gram model compared to the layered softmax used in DeepWalk.

Node2vec[13] is a network embedding method that considers DFS neighborhoods and BFS neighborhoods, it is an extension of DeepWalk.

GraRep[14], Metapath2vec[15] and GCN[16] are also methods of Network Embedding.

C. Adaptive Learning Path Recommendation

The learning path recommendation algorithm is the key to realizing the recommendation of personalized learning paths. It is an interdisciplinary issue, and more and more researchers are trying to work on this issue from different aspects. G. Durand[17], and M. Fiqri[18] proposed methods for constructing a learning path recommendation system based on graph theory. N. Jyothi designed an analysis engine that can help educators identify learners with similar learning styles [19]. H. Supic proposed a reasoning (CBR) model based on historical learner case[20].

In recent years, neural networks have achieved good results in the fields of natural language processing, machine vision, and speech recognition. The artificial neural network model is established according to the learner's learning data, the learning path is recommended, and the weight of the nodes in the network is adjusted according to the difference between the actual learning path and the recommended path. Y. Zhou proposed a new model of learning path recommendation[21], first clustering learners and training long-term short-term memory (LSTM) models to predict learners' learning paths and performance, and then recommend a personalized learning path based on the predicted results. H. Zhu proposed a multi-constrained learning path recommendation algorithm based on different learning scenarios of heterogeneous information networks [22], and the similarity between the learner's self-organizing learning path and the recommended learning path is verified by experiments. C. Krauss proposed a method of recommending learning paths through a knowledge network [23], which at the beginning of the course, the recommended path is also highly correlated.

III. METHOD

Different learners may have different learning effects when the learning paths are the same, so the adaptive learning system should be able to identify learners with the same or similar characteristics， and record the learning effect of historical learners on this path, in order to achieve a better recommendation.

A. Learner Model based on Node2vec

First, we define learners, courses, learners' various attributes and course attributes as nodes in the network. When a learner or course has certain attributes, or the learner takes a course, then there is an edge between the corresponding nodes. So we form a network containing various types of nodes in the learning system. Then we use the node2vec model to learn the expression of each node. The details are as follows:

Node2vec introduces the width first search and the depth first search into the generation process of the random walk sequence by introducing two parameters p and q. Breadth first search (BFS) focuses on neighboring nodes and characterizes a relatively local representation of the network; depth first search (DFS) reflects the homogeneity between nodes at higher levels. BFS can explore the structural properties of graphs, while DFS can explore the similarities between adjacent nodes.

The two parameters p and q in node2vec control the jump probability of the random walk sequence, as shown in Figure 2. Assuming that the edge of the previous step is (t, v), then for different neighbors of node v, node2vec defines the jump probability of different neighbors according to p and q, p controls the probability of jumping to the neighbor of node t, q control the probability of jumping to the non-neighbor of the node t, the specific unnormalized jump probability value is as follows:

$$\alpha_{pq}(t,x) = \begin{cases} \frac{1}{p}, & \text{if } d_{tx} = 0, \\ 1, & \text{if } d_{tx} = 1, \\ \frac{1}{q}, & \text{if } d_{tx} = 2. \end{cases} \quad (1)$$

Where d_{tx} represents the shortest distance between nodes t and x. In order to obtain the optimal values of hyperparameters p and q, node2vec uses a semi-supervised form to determine the most appropriate parameter learning node representation by grid search.

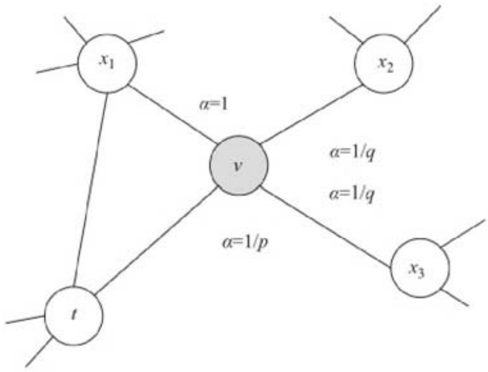

Figure 2. Jump probability of random walk in Node2vec

After the random walk sequence is generated, the sequence is sent to the Skip-Gram model to get the representations of the nodes. Finally, we calculate the cosine similarity of any two learners:

$$sim(u_a, u_b) = \cos\langle u_a, u_b \rangle = \frac{u_a \cdot u_b}{\|u_a\| \times \|u_b\|}, \quad (2)$$

where $\underline{u_a}$ is the representation of learner a, and u_b is the representation of learner b.

B. Generating Course Sequence

The learning record of all learners is expressed in the form of (u, c, q, g), indicating that the learner u took the course c in the quarter q and the grade is g. Then we generate course sequences CS in the form of seq=(c_{pre}, c_{next}, u, g), indicating that the learner u took c_{next} after taking c_{pre}, and the grade of course c_{next} is g. When $c_{pre} = c_0$, the first course that the learner u selects is c_{next}.

Suppose all the behaviors of learner u_1 are recorded as (u_1, c_1, q_1, 0.67), (u_1, c_2, q_2, 0.78), (u_1, c_3, q_2, 0.85), and (u_1. c_4, q_3, 0.77), learner u_1 took two courses in q_2. The course sequences are (c_0, c_1, u_1, 0.67), (c_1, c_2, u_1, 0.78), (c_1, c_3, u_1, 0.85), (c_2, c_4, u_1, 0.77), (c_3, c_4, u_1, 0.77) for u_1.

C. Learning Path Recommendation

Recommend the learning path for the current learner, not only considering the similarity between the learner and other learners, but also the learning effect of the historical learner. Intuitively, assuming that the learning style or learning characteristics of u_a and u_b are quite different, then u_b learning according to the learning path of u_a does not necessarily achieve good learning effects; in addition, if the learning style or learning characteristics of u_a and u_b are similar, but the learning effect of u_a on a particular learning path is not good, then the learning effect of u_b on the learning path is probably not good. Therefore, we use the similarity of learners and the learning effect of historical learners to continuously recommend the unselected courses with the highest scores for the current learners to form a learning path. The scores for the courses are calculated as follows:

$$score(c_x, c_y, u_i) = \sum_{seq \in S} sim(u_i, u) \cdot g, \quad (3)$$

where u_i is the current learner, c_x is the current course of study, $c_y \in C$ is the candidate course, $C=\{c_{next} : seq \in CS \wedge c_{pre}=c_x \wedge c_{next} \in NS\}$, NS is a set of courses that learner u_i has not selected, and $S=\{seq : seq \in CS \wedge c_{pre}=c_x \wedge c_{next}=c_y\}$.

We choose the course with the highest score to add to the learning path, and then continue to select other unselected courses, repeat the process until the path length reaches the set maximum or there is no unselected course.

IV. EXPERIMENTS

In this section, we will introduce the dataset used in the experiment, and analyze the experimental results.

A. Dataset

The Canvas Network is an open online course platform under the American education technology company Instructure. In order to enable researchers to better understand and describe the MOOC phenomenon and online learning behavior and curriculum design features, Canvas Network published a dataset from January 2014 to September 2015 in March 2016.

The dataset includes a total of 32,783 learners and 238 courses in 10 disciplines on the Canvas Network platform. Among the learners, the number of learners with a master's degree is the highest, accounting for 36.51%, followed by bachelor's degree, accounting for 22.81%. Most of the 238 courses are offered in 1-2 quarters, with only 26 courses spanning 3 quarters and above, and the length of the course is concentrated between 35 and 65 days.

These data include 325,199 records, each of which represents one learner's learning behavior in one course. The data set provides a total of 26 fields, which can be divided into four types of data including courses and learners, which are course information, learner basic information, learner learning intentions, and learner education information.

B. Experimental Setup

In node2vec, the dimension of the node feature vector is 128, and the sequence length of the random walk is 50, and grid search over p, $q \in \{0.25, 0.50, 1, 2, 4\}$.

C. Result Analysis

We use data from the first quarter of 2014 to the second quarter of 2015 as training data, and the third quarter of 2015 as test data. Then we compare the average scores of the learners in the four cases, the first is the average grade without any model (WM); the second is the model that introduces the learning similarity(LS), in which the calculation of the scores of the candidate courses is no longer multiplied by grade; the third is the model that introduces the learning effects (LE), in which the calculation of the scores of the candidate courses is no longer multiplied by learner similarity, the fourth is a model that introduces both learner similarity and learning effects(LSLE). For the latter two cases, our test method is to select learners whose elective courses are consistent with the model recommendation results and calculate the average grade. The experimental results are shown in Table I.

TABLE I. EXPERIMENTAL RESULTS

Models	Average Grade
WM	0.4321
LS	0.4732
LE	0.4946
LSLE	**0.5131**

Comparing the experimental results in Table I shows that our method integrates the similarity of learners in an effective way, and the recommendation results can improve the learning effect of learners.

V. CONCLUSION

In this paper, we propose a new method that combines learner similarity and learning effects to recommend learning paths. We evaluated our method on the Canvas Network dataset. The similarity between current learners and other learners and the learning effects of historical learners can improve the recommendation effect. The experimental results show that our method helps to improve the learning effect compared with the learner's own choice of learning path.

REFERENCES

[1] T. I. Wang, K. T. Wang, Y. M. Huang. Using a style-based ant colony system for adaptive learning[J]. Expert Systems with Applications, 2008, 34(4): 2449-2464.

[2] Y. J. Yang, C. Wu. An attribute-based ant colony system for adaptive learning object recommendation[J]. Expert Systems with Applications, 2009,36(2): 3034-3047.

[3] C. M. Chen, H. M. Lee, Y. H. Chen. Personalized e-learning system using item response theory[J]. Computers & Education, 2005, 44(3): 237-255.

[4] A. Baylari, G. A. Montazer. Design a personalized e-learning system based on item response theory and artificial neural network approach[J]. Expert Systems with Applications, 2009,36(4): 8013-8021.

[5] V. Carchiolo, A. Longheu, M. Malgeri. Reliable peers and useful resources: Searching for the best personalised learning path in a trust- and recommendation-aware environment[J]. Information Sciences, 2010,180(10): 1893-1907.

[6] B. H. Li, Z. G. Li. Improved algorithm based on mutual information for learning Bayesian network structures in the space of equivalence classes[J]. Multimedia Tools and Applications, 2012,60(1): 129-137.

[7] P. Brusilovsky, "Adaptive hypermedia," in Ten Year Anniversary Issue, Kluwer Academic Publishers, 2001, pp. 87–110.

[8] A. Kavcic, "The role of user models in adaptive hypermedia systems," in Electrotechnical Conference, 2000. MELECON 2000. 10th Mediterranean, 2000, vol. 1, pp. 119–122 vol.1.

[9] S. Graf, Kinshuk, and L. Tzu-Chien, "Supporting teachers in identifying students' learning styles in learning management systems: an automatic student modelling approach," Educational Technology & Society, vol. 12, no. 4, pp. 3–14, 2009.

[10] E. Siakas and A. . Economides, "Adaptive learning: mapping personality types to learning styles," presented at the INSPIRE, Thessaloniki, 2012.

[11] B. Perozzi, R. Al-Rfou, and S. Skiena, "DeepWalk: online learning of social representations," in The 20th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD '14, New York, NY, USA - August 24 - 27, 2014, 2014, pp. 701–710.

[12] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, and Q. Mei, "LINE: Large-scale Information Network Embedding," in Proceedings of the 24th International Conference on World Wide Web, WWW 2015, Florence, Italy, May 18-22, 2015, 2015, pp. 1067–1077.

[13] A. Grover and J. Leskovec, "node2vec: Scalable Feature Learning for Networks," in Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Francisco, CA, USA, August 13-17, 2016, 2016, pp. 855–864.

[14] S. Cao, W. Lu, and Q. Xu, "GraRep: Learning Graph Representations with Global Structural Information," in Proceedings of the 24th ACM International Conference on Information and Knowledge Management, CIKM 2015, Melbourne, VIC, Australia, October 19 - 23, 2015, 2015, pp. 891–900.

[15] Y. Dong, N. V Chawla, and A. Swami, "metapath2vec: Scalable Representation Learning for Heterogeneous Networks," in Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax, NS, Canada, August 13 - 17, 2017, 2017, pp. 135–144.

[16] M. Niepert, M. Ahmed, and K. Kutzkov, "Learning Convolutional Neural Networks for Graphs," in Proceedings of the 33nd International Conference on Machine Learning, ICML 2016, New York City, NY, USA, June 19-24, 2016, 2016, pp. 2014–2023.

[17] G. Durand, N. Belacel, and F. LaPlante, "Graph theory based model for learning path recommendation," Inf. Sci., vol. 251, pp. 10–21, 2013.

[18] M. Fiqri and D. Nurjanah, "Graph-based domain model for adaptive learning path recommendation," in 2017 IEEE Global Engineering Education Conference, EDUCON 2017, Athens, Greece, April 25-28, 2017, 2017, pp. 375–380.

[19] N. Jyothi, K. Bhan, U. Mothukuri, S. Jain, and D. Jain, "A Recommender System Assisting Instructor in Building Learning Path for Personalized Learning System," in 2012 IEEE Fourth International Conference on Technology for Education, T4E 2012, Hyderabad, India, July 18-20, 2012, 2012, pp. 228–230.

[20] H. Supic, "Case-Based Reasoning Model for Personalized Learning Path Recommendation in Example-Based Learning Activities," in 27th IEEE International Conference on Enabling Technologies: Infrastructure for Collaborative Enterprises, WETICE 2018, Paris, France, June 27-29, 2018, 2018, pp. 175–178.

[21] Y. Zhou, C. Huang, Q. Hu, J. Zhu, and Y. Tang, "Personalized learning full-path recommendation model based on LSTM neural networks," Inf. Sci., vol. 444, pp. 135–152, 2018.

[22] H. Zhu et al., "A multi-constraint learning path recommendation algorithm based on knowledge map," Knowl.-Based Syst., vol. 143, pp. 102–114, 2018.

[23] C. Krauss, A. Salzmann, and A. Merceron, "Branched Learning Paths for the Recommendation of Personalized Sequences of Course Items," in Proceedings der Pre-Conference-Workshops der 16. E-Learning Fachtagung Informatik co-located with 16th e-Learning Conference of the German Computer Society (DeLFI 2018), Frankfurt, Germany, September 10, 2018., 2018, vol. 2250.