Information Processing and Management 61 (2024) 103536

Contents lists available at ScienceDirect

Information Processing and Management

journal homepage: www.elsevier.com/locate/ipm

Quantification and prediction of engagement: Applied to personalized course recommendation to reduce dropout in MOOCs

Shu Li [a,b,c,f,1], Yuan Zhao [d,1], Longjiang Guo [a,b,c,f,g,*], Meirui Ren [a,b,c,f,g], Jin Li [a,b,c,f,g], Lichen Zhang [a,b,c,f,g], Keqin Li [e]

[a] Key Laboratory of Intelligent Computing and Service Technology for Folk Song, Ministry of Culture and Tourism, Xi'an 710119, China
[b] School of Computer Science, Shaanxi Normal University, Xi'an 710119, China
[c] Key Laboratory of Modern Teaching Technology, Ministry of Education, Xi'an 710062, China
[d] College of Computer Science and Engineering, Northwest Normal University, Lanzhou, China
[e] Department of Computer Science, State University of New York, New Paltz, NY 12561, United States of America
[f] Engineering Laboratory of Teaching Information Technology of Shaanxi Province, Xi'an 710119, China
[g] Xi'an Key Laboratory of Culture Tourism Resources Development and Utilization, Xi'an 710062, China

ARTICLE INFO

Dataset link: http://moocdata.cn/data/user-activity

Keywords:
Engagement
MOOC
Artificial intelligence in education
Recommendation mechanism
Reduce dropout

ABSTRACT

MOOCs (Massive Open Online Courses) offer tens of thousands of courses and attract hundreds of millions of online learners. After years of development, these platforms have accumulated a large number of learning action data. These data imply learners' engagement with the enrolled courses, reflecting whether learners are willing to spend time on the courses. Meanwhile, the MOOC dropout rate remains chronically high. This paper experimentally finds that using the relationship between engagement and dropout rate can help MOOC platforms develop effective methods to reduce the dropout rate. Firstly, this paper proposes a new Quantified Engagement method named QE by using learning action data and learning duration to quantify learners' engagement in enrolled courses. Next, an Engagement Neural Network prediction model named ENN is proposed to predict learners' engagement in unenrolled courses. Then, applying QE, the predicted engagement by ENN, and the aforementioned relationship to personalized course recommendations to learners, ensuring that the recommended courses are likely to be completed by learners as much as possible, thus effectively reducing the dropout rate. Finally, the proposed method is evaluated on two large real-world datasets in XuetangX and KDDCUP. The RMSE and MAE of ENN are 0.1066 and 0.0727 on XuetangX and 0.062411 and 0.039621 on KDDCUP, respectively. Dropout rates were reduced by 46.99% and 10.34%, respectively, when 5% of the courses were recommended. These results demonstrate that the quantification method of engagement is valid, applying predicted engagement to personalized course recommendations and reducing dropout rates is available.

* Corresponding author at: School of Computer Science, Shaanxi Normal University, Xi'an 710119, China.
E-mail addresses: ls980108@163.com (S. Li), zhaoyuan2233@163.com (Y. Zhao), longjiangguo@snnu.edu.cn (L. Guo), meiruiren@snnu.edu.cn (M. Ren), jin.li@snnu.edu.cn (J. Li), zhanglichen@snnu.edu.cn (L. Zhang), lik@newpaltz.edu (K. Li).
[1] The first two authors contributed equally.

https://doi.org/10.1016/j.ipm.2023.103536
Received 13 April 2023; Received in revised form 8 September 2023; Accepted 15 October 2023
Available online 24 October 2023
0306-4573/

1. Introduction

With the help of the Internet, Artificial Intelligence and education begin to integrate deeply, and various MOOC (Massive Open Online Course) platforms have emerged and developed rapidly, including Coursera,[2] edX,[3] Udacity,[4] XuetangX,[5] FutureLearn,[6] etc. After years of development, these platforms have accumulated a large number of users with their high-quality and massive learning resources. Among them, XuetangX is one of the largest MOOC platforms in China, offering tens of thousands of courses and attracting hundreds of millions of users to learn (Feng, Tang, & Liu, 2019). Learners generate a huge amount of learning action data during the learning process, which implies their engagement with the course, reflecting whether they are willing to spend time on the course. If engagement can be quantified, it can help MOOC platforms identify learners' willingness to continue learning in time, so that they can develop strategies to reduce dropout rates.

Quantification and prediction of learners' engagement have theoretical and practical implications. According to People's Daily's[7] latest statistical analysis, as of November 2022, the number of courses in MOOC exceeded 61,900, with 979 million learners. One of the more serious problems on MOOC platforms is the persistently high dropout rate (Borrella, Caballero-Caballero, & Ponce-Cueto, 2022; Cheng, Nunes, & Manrique, 2022; D'Aniello, de Falco, Gaeta, & Lepore, 2020; Feng et al., 2019; Molina, Obando, Bastidas, & Mosquera, 2022). Learners stop learning halfway through the course, and a large number of learning resources are untouched by the learners, which leads to a waste of platform resources. Quantifying and predicting learners' engagement can help MOOC platforms to observe learners' learning status and determine whether they are willing to spend time on the enrolled courses. Also, if the relationship between engagement and dropout rate can be analyzed as well as predicting learners' willingness to study unenrolled courses, it will also help develop targeted approaches to reduce dropout rates. In addition, quantified and predicted learners' engagement can reflect the quality of course design. If the majority of learners have a low level of engagement in the course, it indicates a lack of interest in the course by the majority of learners, which is a further indication that the course may not be popular with the majority of learners. At this point, the MOOC platforms go to work to remind course designers to improve their course design to attract more learners to the course.

The quantification and prediction of learners' engagement studied in this paper are fundamentally different from the existing engagement-related work. Several researchers have already started to focus on engagement. Mehta, Prasad, Saurav, Saini, and Singh (2022) introduced a three-dimensional DenseNet SelfAttention neural network for automatically detecting student engagement on e-learning platforms. Cole, Lennon, and Weber (2021) correlated self-reports of student engagement intentions with student behavior tracking data in an online course management system to assess and measure student engagement intentions in online courses. O'Brien, Arguello, and Capra (2020) captured self-reported task perceptions and recorded search behaviors to discover task topics influencing user engagement. Song, Rice, and Oh (2019) analyzed the frequency and length of course visits, discussion board posts, and final scores to obtain learners' engagement and found it to be strongly correlated with their course scores. Soffer and Cohen (2019) explored the characteristics of students' engagement on online courses and their impact on academic performance and attempts to predict learners' completion. However, these research works only analyzed learners' engagement and have not been able to predict it in unenrolled courses, link it to reduced dropout rates, or develop specific methods to reduce dropout rates.

This paper's approach to reducing dropout rates based on quantification and prediction of engagement is also quite different from existing approaches to reducing dropout rates. To date, few studies have explored how to reduce dropout rates in MOOC platforms. Most studies around dropout rates have explored the main factors that lead learners to drop out and proposed predictive models to identify learners at risk of dropping out (Cheng et al., 2022; Drousiotis, Pentaliotis, Shi, & Cristea, 2021; Feng et al., 2019; Mogavi, Ma, & Hui, 2021). This paper designs a strategy to reduce the dropout rate. The core idea of this strategy entails utilizing the correlation between engagement quantified by QE and dropout rates, as found in this paper, to elucidate that learners have the potential to complete courses with high engagement. Additionally, it aims to recommend unenrolled courses with high engagement to learners for course selection. This strategy ensures that the recommended courses are more likely to be completed by learners as much as possible, and it is fundamentally different from existing course recommendations. Existing course recommendation methods focus on whether learners will click, register, or be interested in the recommended courses (Lin et al., 2022; Sakboonyarat, Siriporn, Tantatsanawong, & Panjai, 2019; Wang, Ma, et al., 2022; Yang & Cai, 2022), but do not consider whether learners can complete the recommended courses. Therefore, existing course recommendation methods do not provide in-depth research on how to reduce dropout rates.

This paper proposes a new method to Quantify Engagement (QE) by using learning action data and learning duration to quantify learners' engagement in enrolled courses, to determine whether they are willing to spend time in enrolled courses, and experimentally find that learners with high quantified engagement are more likely to complete enrolled courses. Then, an Engagement Neural Network (ENN) prediction model is proposed to predict learners' engagement in unenrolled courses by using the idea of matrix decomposition and combining it with deep learning, which will help MOOC platforms to measure their willingness to continue learning in unenrolled courses. Finally, using the engagement of unenrolled courses predicted by ENN as well as the relationship between engagement and dropout rate, personalized course recommendations are made to learners to ensure that the

[2] https://www.coursera.org/.
[3] https://www.edx.org/.
[4] https://www.udacity.com/.
[5] https://www.xuetangx.com/.
[6] https://www.futurelearn.com/.
[7] http://paper.people.com.cn/rmrb/html/2023-02/12/nw.D110000renmrb_20230212_1-05.htm.

recommended courses are likely to be completed by learners, thus effectively reducing the dropout rate of the MOOC platforms. The main contributions are summarized as follows.

- **Giving Quantitation Engagement in the Enrolled Courses and Finding the Relationship between Quantified Engagement and Dropout Rate**. That aims to quantify whether learners are willing to spend time on enrolled courses using record of learning activities and learning duration, which is defined as engagement. Then, this paper also finds the dropout rates of learners reduce gradually with the increase in their quantified engagement. This paper defines this quantification method as QE.
- **Proposing Prediction of Engagement in the Unenrolled Courses using Neural Network and Finding the Relationship between predicted Engagement and Dropout Rate.** It aims to use the deep learning method to complete the matrix by considering the interaction between learners and courses and their engagement in the enrolled courses, to predict the engagement in unenrolled courses. By finding the relationship between predicted engagement and dropout rate, this paper observes that learners with lower predicted engagement were more likely to dropout. This paper defines the prediction model as ENN.
- **Presenting Personalized Course Recommendations by Prediction of Engagement to Reduce Dropout Rate.** In this paper, the utilization of ENN to predict engagement in unenrolled courses, combined with the relationship between predicted engagement and dropout rate, facilitates the generation of personalized course recommendations for learners. The objective is to enhance the likelihood of learners completing the recommended unenrolled courses. Comprehensive experiments show that this will be more beneficial to reduce the dropout rate of MOOC platforms.
- **Making Comprehensive evaluation.** Thorough experiments on real-world datasets in XuetangX and KDDCUP demonstrate that the ENN has good predictive capabilities, using engagement to recommend courses for learners can achieve better recommendation results and makes the learners more likely to complete the recommended courses, which will greatly reduce learners' dropout rates. The RMSE and MAE of the ENN model were 0.1066 and 0.0727 on the XuetangX dataset, and 0.0624 and 0.0396 on the KDDCUP dataset, respectively. For top-K recommendations, the Hit ratio of our experiments improved by an average of 2.75%, the Recall by an average of 3%, the Precision by an average of 3.38%, and F1-score by an average of 3.62% on two datasets. When recommending 5% of courses, the dropout rate on the MOOC platforms is reduced averagely by 28.67% on two datasets. The source code for this paper has been published.[8]

The remainder of the paper is organized as follows. Related work is introduced in Section 2. Section 3 describes the problem formulation. Section 4 proposes a quantitative method of QE. In Section 5, a predictive model of ENN is proposed. Section 6 details how to use predicted engagement to personalize course recommendations. Section 7 conducts extensive experiments to verify the validity of the proposed model through experimental results. Finally, Section 8 gives a summary and outlines future work.

2. Related work

This section investigates and studies the work related to learner engagement, dropout rate and course recommendation, the details are as follows.

2.1. Learner engagement

Ma, Han, Yang, and Cheng (2015) mentioned that student engagement usually refers to the amount of time students invest in their academic experience. Plak, van Klaveren, and Cornelisz (2023) wrote engagement in educational activities is an important prerequisite for academic success. Mehta et al. (2022) introduced a three-dimensional DenseNet SelfAttention neural network for automatically detecting student engagement on e-learning platforms. Cole et al. (2021) correlated self-reports of student engagement intentions with student behavior tracking data in an online course management system to assess and measure student engagement intentions in online courses. Soffer and Cohen (2019) explored the characteristics of students' engagement in online courses and their impact on academic performance and attempted to predict learner completion. Almutairi and White (2018) used a statistical approach to measure learners' engagement by designing a questionnaire. The questionnaire approach is not suitable for the automatic quantification of learner engagement in MOOC platforms with a large number of learners and a large number of courses. Raguro, Lagman, Abad, and Ong (2022) used a decision tree and K-Means algorithm to confirm the existence of a strong relationship between student behavior and academic performance. Huang, Lu, and Yang (2023) mentioned that personalized video recommendations can significantly improve academic performance and engagement of students with moderate motivation levels.

In the above studies on engagement, many works have mentioned that engagement is closely related to learners' academic completion, but no work can predict learners' engagement in unenrolled courses, nor link learner engagement with reducing the dropout rate. This paper proposes a method to quantify learners' engagement in enrolled courses and uses the ENN model to predict learners' engagement in unenrolled courses. Finally, personalized course recommendations are proposed to reduce dropout rates.

[8] https://github.com/lishushushushu/PCRM.git/.

2.2. Dropout rate

Feng et al. (2019) built a CFIN model to predict user dropout rates in MOOCs. Lin, Sun, et al. (2021) designed a two-tower framework to predict whether students will drop out this semester. Xing (2019) developed a deep learning algorithm that uses a weekly time prediction mechanism to build a dropout model. Cheng et al. (2022) used students' academic data to predict whether these students will drop out of school in the next semester. There are some other works (Drousiotis et al., 2021; Hancox & Relton, 2022; Mogavi et al., 2021; Zhang & Ma, 2022) that propose methods to do dropout prediction.

Molina et al. (2022) referred to conducting content recommendations and personalization to reduce learner dropout rates. D'Aniello et al. (2020) proposed a feedback generation approach aimed at increasing learner motivation and engagement to reduce student dropout rates. Borrella et al. (2022) proposed a framework to design interventions (using individual factors as targets and institutional factors as levers) and provides effective methods to reduce dropout (focusing placed on content and instructional design).

Most of the above works on dropout rates explored the main factors that cause learners to drop out and proposed predictive models to identify at-risk learners. However, there are few studies on reducing the dropout rate. In this paper, the method of course recommendation is adopted to develop personalized learning courses for learners, effectively reducing the dropout rate of MOOC platforms.

2.3. Course recommendation

MOOC resources are numerous and face serious information overload problems, so recommendation system is gradually popular. It can provide learners with personalized information to meet their learning preferences.

Wang, Ma, et al. (2022) developed a hyperedge-based graph neural network (HGNN), which treated learners as the sets of courses in a hypergraph, and considered courses' long-term and short-term sequential relationships to make course recommendation, the recommendation aims specifically at the course in which the learner will be enrolled in the upcoming stage. Sakboonyarat et al. (2019) used deep learning models with multi-layer perceptron architecture suitable for large amounts of data for course recommendation, to enable the recommended courses to be enrolled by learners. In addition, there are other works that also judge the recommendation effect according to whether learners enroll for the recommended course (Jung, Jang, Kim, & Kim, 2022; Lin et al., 2022; Obeidat, Duwairi, & Al-Aiad, 2019; Shao, Guo, & Pardos, 2021), but these works fail to pay attention to whether learners will complete the course.

Zhang, Shen, Yi, Wang, and Feng (2023) proposed a high-performance course recommendation model that uses a heterogeneous graph describing the relationship between courses and facts to automatically and iteratively estimate the click probability of learners. Yang and Cai (2022) took a knowledge graph as an auxiliary information source for collaborative filtering and proposed an end-to-end framework using a knowledge graph to enrich the semantics of item representation. Apply the deep matrix factorization model together with the improved loss function to the course proposal. Other related work is as follows (Ban et al., 2022; Ma, Wang, Chen, & Shen, 2021; Yao, Sun, & Hu, 2020), but all of these work only focused on whether learners would click on the recommended course, not whether learners would complete the course.

Xu, Jia, Shi, and Zhang (2021) tried to use an algorithm combining knowledge graph and collaborative filtering for course recommendation. Wang, Zhu, et al. (2022) designed a framework of demand-aware Collaborative Bayesian Variational Network (DCBVN) and Demand-aware Collaborative Ability Attention Network (DCCAN) for course recommendation. But these works focus on whether learners respond to the recommended course, not whether learners complete the course.

As we can see, most of the hit standards of course recommendation only focus on whether learners will click on, register for, or be interested in the recommended course, but do not pay much attention to whether learners can complete the recommended course. This paper takes this into account by quantifying and predicting engagement, and recommending courses based on the relationship between engagement and dropout rates, so that learners are not only interested in the recommended courses but also complete the recommended courses, which are used as the recommendation hit standard in this paper, it reduces the dropout rate of MOOC platforms. At the same time, due to a large number of MOOC resources, the recommendation systems of online platforms often have the problem of sparse rating data (Li et al., 2021). In this paper, not only the engagement matrix but also the rich behavioral information is considered in the recommendation.

2.4. Summary

To sum up, quantification and prediction of learners' engagement have theoretical and practical implications. The quantification and prediction of learners' engagement studied in this paper are quite different from the existing engagement-related work, and the approach to reducing dropout rates differs significantly from existing methods. Specifically, existing work on learner engagement has focused solely on analyzing the correlation between engagement and grades in enrolled courses. However, this paper not only quantifies learners' engagement in enrolled courses by utilizing their action data and learning duration but also establishes a relationship between engagement and dropout rates through experimental analysis. Furthermore, an ENN model is proposed for predicting learners' engagement in unenrolled courses. Unlike previous research, which tends to predict learners' dropout rates in enrolled courses without linking it to engagement, this paper explores how to reduce dropout rates from a personalized recommendation perspective. By recommending courses with higher engagement to learners, this approach aims to increase the likelihood of learners completing the recommended courses, thereby reducing dropout rates.

Table 1
Behavior and Action from XuetangX and KDDCUP dataset.

	XuetangX	KDDCUP
Behaviors	Actions	Behaviors
Video	Seek_video , Play_video, Pause_video, Stop_video, Load_video.	Video
Problem	Problem_check, Problem_save, Problem_check_correct, Problem_check_incorrect.	Problem
Forum	Click_forum, Close_forum.	Forum
Courseware	Click_courseware, Close_courseware.	Wiki
		Discussion

3. Problem formulation

The set of learners on the MOOCs is $U = \{u_1, u_2, \ldots, u_i, \ldots, u_{|U|}\}$. The course set include in the online learning platforms is $C = \{c_1, c_2, \ldots, c_j, \ldots, c_{|C|}\}$. For each u_i and his enrolled courses set can be defined as $C(u_i)^+ = \{c_j \mid u_i \text{ enrolled in } c_j,\ c_j \in C\}$. $D(u_i) = \{c_j \mid u_i \text{ enrolled in } c_j \text{ but dropout},\ c_j \in C(u_i)^+\}$, $C(u_i)^+ - D(u_i) = \{c_j \mid u_i \text{ enrolled in } c_j \text{ and complete},\ c_j \in C(u_i)^+\}$. $C(u_i)^- = \{c_j \mid u_i \text{ unenrolled in } c_j,\ c_j \in C\}$ represents the set of courses unenrolled by u_i, where $C(u_i)^+ \subseteq C$, $C(u_i)^- \subseteq C$. $U(c_j)^+ = \{u_i \mid u_i \text{ enrolled in } c_j,\ u_i \in U\}$ denotes the set of learners enrolled in c_j, $U(c_j)^- = \{u_i \mid u_i \text{ unenrolled in } c_j,\ u_i \in U\}$ denotes the set of learners unenrolled in c_j, where $U(c_j)^+ \subseteq U$, $U(c_j)^- \subseteq U$.

3.1. Fundamental definitions

Definition 1 (*Learning Action Types*)**.** After u_i enrolled in c_j, there will be a lot of learning behaviors, and each type of learning behavior is subdivided into a variety of actions. The set of different types of learning behaviors is denoted by $B\left(u_i, c_j\right) = \{b_{ij}^{(1)}, b_{ij}^{(2)}, \ldots, b_{ij}^{(v)}, \ldots, b_{ij}^{(|B(u_i,\ c_j)|)}\}$, where $b_{ij}^{(v)} \in B\left(u_i, c_j\right)$ represents the vth learning behavior type. $b_{ij}^{(v)} = \{a_{i,j,v}^{(1)}, a_{i,j,v}^{(2)}, \ldots, a_{i,j,v}^{(p)}, \ldots, a_{i,j,v}^{|(b_{ij}^{(v)})|}\}$, $A\left(u_i, c_j, b_v\right) = b_{ij}^{(v)}$ represents the set of actions contained in the vth behavior. $A\left(u_i, c_j\right)$ represents the set of different types of learning actions that arise after u_i enrolled in c_j, where $A\left(u_i, c_j, b_v\right) \in A\left(u_i, c_j\right)$.

$B\left(u_i, C(u_i)^+\right) = \bigcup_{u_i \in U,\ c_j \in C(u_i)^+} B\left(u_i,\ c_j\right)$ denotes the set of learning behavior types of u_i in all of his enrolled courses, $B = \bigcup_{u_i \in U,\ c_j \in C} B\left(u_i,\ c_j\right)$ represents all possible types set of learning behaviors. The learning action types set of u_i in all of his enrolled courses can be denoted as $A(u_i, C(u_i)^+) = \bigcup_{u_i \in U,\ c_j \in C(u_i)^+} A\left(u_i,\ c_j\right)$, $A = \bigcup_{u_i \in U,\ c_j \in C} A\left(u_i,\ c_j\right)$ denotes all possible types set of learning actions.

Example 1. Table 1 shows detailed data on behaviors and actions in both the XuetangX and KDDCUP datasets. Among them, only the data for behavior is available in KDDCUP. However, we can understand that the learning behavior data in KDDCUP is similar to the learning action data in XuetangX. For the convenience of writing the paper, all the methods described later are based on the data in the XuetangX dataset, and the data in the KDDCUP dataset were processed similarly. On the XuetangX dataset, $B = \{Video, Problem, Forum, Courseware\}$, $A = \{Play\_video,\ Problem\_check\_correct,\ \ldots,\ Close\_courseware\}$.

Definition 2 (*Learning Action Type Clicks*)**.** Learning action type clicks set can be regarded as the number of times that learners operate a certain action in the course, and each click will be recorded. $N\left(u_i, A\left(u_i, c_j, b_v\right)\right) = \{n_{i,j,v}^{(1)}, n_{i,j,v}^{(2)}, \ldots, n_{i,j,v}^{(p)}, \ldots, n_{i,j,v}^{(|b_{ij}^{(v)}|)}\}$ represents the clicks set of each action on the vth behavior arising after u_i enrolled in c_j. The clicks set for different types of learning actions about u_i on c_j is denoted by $N\left(u_i, A\left(u_i, c_j\right)\right)$, it can be expressed as $N\left(u_i, A\left(u_i, c_j\right)\right) = \{n_{ij}^{(1)}, n_{ij}^{(2)}, \ldots, n_{ij}^{(p)}, \ldots, n_{ij}^{(|A(u_i, c_j)|)}\}$, where $n_{ij}^{(p)}$ is pth learning action type clicks. $N\left(u_i, A\left(u_i, C(u_i)^+\right)\right)$ denotes the clicks set of each actions about u_i in enrolled courses.

Definition 3 (*Information Matrices* **UC**, **UA** *and* **CA**)**.**Different learners show different learning actions in the course, this paper aggregate the learning actions of learners and courses respectively and obtain three relationship matrices. **UC** denotes the incidence matrix of learners and courses, **UC**$= (uc_{ij})_{|U| \times |C|}$, if $c_j \in C(u_i)^+$, $uc_{ij} = 1$, otherwise $uc_{ij} = 0$. **UA** denotes the average aggregation matrix of courses, **UA**$= (ua_{ip})_{|U| \times |A|}$, ua_{ip} represents the average of pth learning action type clicks for courses enrolled by u_i, $ua_{ip} = \frac{1}{|C(u_i)^+|} \sum_{c_j \in\ C(u_i)^+} n_{ij}^{(p)}$. **CA** denotes the average aggregation matrix of learners, **CA** $= (ca_{jp})_{|C| \times |A|}$, $ca_{jp} = \frac{1}{|U(c_j)^+|} \sum_{u_i \in\ \mathrm{U}(c_j)^+} n_{ij}^{(p)}$ represents the average of pth learning action type clicks for all learners enrolled in c_j.

Example 2. In Fig. 1,this paper analyzes the XuetangX and KDDCUP dataset, learning activity records for learners and courses are aggregated separately to obtain three matrices. Fig. 1(a) is the incidence matrix, Figs. 1(b) and 1(c) are the average aggregation matrix of courses and the average aggregation matrix of learners, respectively. It is worth noting that since there is no action data in the KDDCUP dataset, the behavior data is used when building the **UA** and **CA**.

Definition 4 (*Learning Duration*)**.** Learning Duration represents the amount of time that a learner spends learning a certain course. t_{ij} denotes the learning duration that u_i spend on enrolled course c_j. The set of learning duration for u_i in enrolled courses can be expressed as $T(u_i,\ C(u_i)^+)$= $\{t_{ij} \mid u_i \text{ enrolled in } c_j,\ c_j \in C(u_i)^+\}$.

UC

	c_1	c_2	$\cdots$	c_n
u_1	1	0	$\cdots$	1
u_2	0	1	$\cdots$	0
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$
u_m	1	0	$\cdots$	0

(a) The incidence matrix of learners and courses

UA

	a_1	a_2	$\cdots$	a_t
u_1	0.2	0.3	$\cdots$	0.4
u_2	0.5	0.8	$\cdots$	0.3
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$
u_m	0.6	0.3	$\cdots$	0.1

(b) The average aggregation matrix of courses

CA

	a_1	a_2	$\cdots$	a_t
c_1	0.1	0.4	$\cdots$	0.2
c_2	0.4	0.8	$\cdots$	0.01
$\cdots$	$\cdots$	$\cdots$	$\cdots$	$\cdots$
c_n	0.3	0.1	$\cdots$	0.9

(c) The average aggregation matrix of learners

Fig. 1. The information matrices **UC**, **UA** and **CA**.

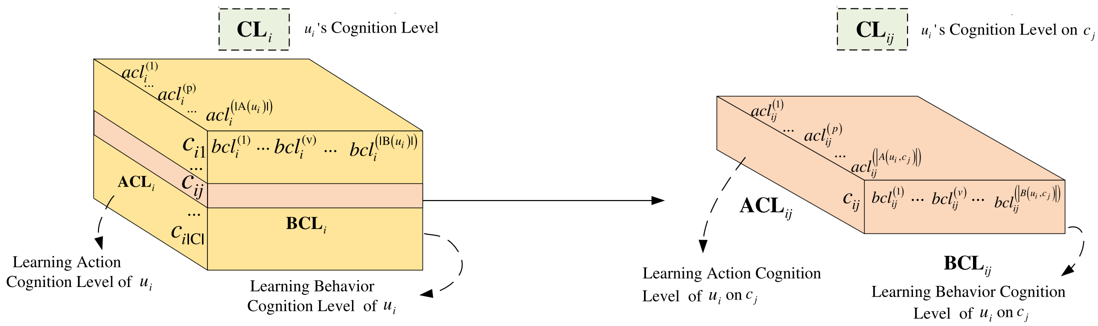


Fig. 2. Diagram of learners' cognition level.

Definition 5 (*Learners' Cognition Level*)**.** The cognition level of u_i in all courses is denoted by $\mathbf{C}L_i \in \mathbb{R}^{|A(u_i,C(u_i)^+)|\times|B(u_i,C(u_i)^+)|\times|C(u_i)^+|}$, $\mathbb{R}$ is real numbers set. $\mathbf{ACL}_i \in \mathbb{R}^{|C(u_i)^+|\times|A(u_i,C(u_i)^+)|}$ denotes the learning action cognition level of u_i in all enrolled courses, $\mathbf{ACL}_{ij} \in \mathbb{R}^{1\times|A(u_i,c_j)|}$ represents the learning action cognition level of u_i on c_j, where $\mathbf{ACL}_{ij} = [acl_{ij}^{(1)},\ldots,acl_{ij}^{(p)},\ldots,acl_{ij}^{(|A(u_i,c_j)|)}]$. The learning behavior cognition level of u_i on all enrolled courses can be expressed as $\mathbf{BCL}_i \in \mathbb{R}^{|C(u_i)^+|\times|B(u_i,C(u_i)^+)|}$, the learning behavior cognition level of each behavior of u_i on c_j can be expressed as $\mathbf{BCL}_{ij} = [bcl_{ij}^{(1)},\ldots,bcl_{ij}^{(v)}\ldots,bcl_{ij}^{(|B(u_i,c_j)|)}]$.

Example 3. Fig. 2 gives a schematic representation of the u_i's cognition level so that the reader can more clearly visualize the representation, where the cognition level of learners can be divided into learning behavior cognition level $\mathbf{BCL}_i$ and learning action cognition level $\mathbf{ACL}_i$. In the following content, this paper will specifically explain how to calculate the learning action cognition level $\mathbf{ACL}_i$, which is calculated in a similar way to the learning behavior cognition level $\mathbf{BCL}_i$.

Definition 6 (*Engagement*)**.** The engagement reflects how active the learner on all courses, and since there is no learning action data in the KDDCUP dataset, the learning behavior cognition level is further quantified as engagement. And learning action cognition level is further quantified as engagement in the XuetangX dataset. It can be defined as $\mathbf{X^A} = (x_{ij})_{|U|\times|C|}$. $\mathbf{X^A}_i = [x_{i1}, x_{i2},\ldots,x_{ij},\ldots,x_{i|C|}]$ represents the engagement of u_i on all courses, where $0 \le x_{ij} \le 1$. The closer x_{ij} is to 1, the higher u_i's engagement in c_j.

The engagement of the u_i in all enrolled course can be expressed as $\mathbf{X^A}_i{}^+ \in \mathbb{R}^{1\times|C(u_i)^+|}$, it can be quantified by QE, which is described in the Section of Quantitation of Engagement in Enrolled Courses.

$\mathbf{X^A}_i{}^- \in \mathbb{R}^{1\times|C(u_i)^-|}$ represents the engagement for u_i in unenrolled courses, where $x_{ij}^- \in \mathbf{X^A}_i^-$, x_{ij}^- also has a potential value between 0 and 1, it can be predicted by the ENN, which is described in the Section of Prediction of Engagement in Unenrolled Courses using Neural Network.

3.2. Research questions

In order to solve the problems presented in this paper, the following three Research Questions (RQ) need to be solved. They are the quantification of engagement, the prediction of engagement, and the personalized course recommendation by prediction of engagement to reduce dropout rate.

RQ 1. How to quantify u_i's engagement in enrolled courses using learning action data and learning duration? The task can be described as: Given $N\left(u_i, A\left(u_i, C(u_i)^+\right)\right)$ and $T(u_i, C(u_i)^+)$, how to design the QE method to quantify u_i's engagement in enrolled courses?

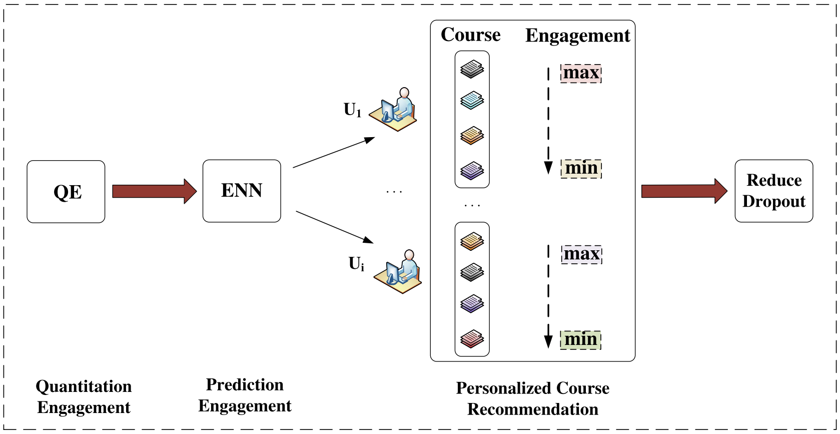


Fig. 3. The overall model architecture.

Table 2
Main symbols and descriptions in this paper.

Symbol	Description
U	Learner set, U={u_1, u_2, …, u_i, …, $u_{\|U\|}$}, where $u_i \in U$.
C	Course set, C={c_1, c_2, …, c_j, …, $c_{\|C\|}$}, where $c_j \in C$.
$A\left(u_i, c_j\right)$	The set of different types of learning actions of u_i on c_j.
$N\left(u_i, A\left(u_i, c_j\right)\right)$	The clicks set for different types of learning actions arising after u_i enrolled in c_j, $N\left(u_i, A\left(u_i, c_j\right)\right) = \{n_{ij}^{(1)}, n_{ij}^{(2)}, \ldots, n_{ij}^{(p)}, \ldots, n_{ij}^{(\|A(u_i,c_j)\|)}\}$.
CA	The average aggregation matrix of learners, $\mathbf{CA} = (ca_{jp})_{\|C\|\times\|A\|}$, ca_{jp} represents the average of pth learning action type clicks for all learners enrolled in c_j.
UC	The incidence matrix of learners and courses, **UC**= $(uc_{ij})_{\|U\|\times\|C\|}$, uc_{ij} represents that whether the learner u_i will chooses course c_j, if $c_j \in C(u_i)^+$, $uc_{ij} = 1$, otherwise $uc_{ij} = 0$.
UA	The average aggregation matrix of courses, $\mathbf{UA} = (ua_{ip})_{\|U\|\times\|A\|}$, ua_{ip} represents the average of pth learning action type clicks for courses enrolled by u_i.
$T(u_i, C(u_i)^+)$	It represents the amount of time that a learner spends learning a certain course. $T(u_i, C(u_i)^+)$= {t_{ij} | u_i enrolled in c_j, $c_j \in C(u_i)^+$ }.
$\mathbf{CL}_i$	u_i's cognition level on all courses, where $\mathbf{CL}_i \in \mathbb{R}^{\|A(u_i,C(u_i)^+)\|\times\|B(u_i,C(u_i)^+)\|\times\|C(u_i)^+\|}$.
$\mathbf{BCL}_i$	The learning behavior cognition level of $u_i$in enrolled courses, where $\mathbf{BCL}_i \in \mathbb{R}^{\|C(u_i)^+\|\times\|B(u_i,C(u_i)^+)\|}$.
$\mathbf{ACL}_i$	The learning action cognition level of u_i in enrolled courses, where $\mathbf{ACL}_i \in \mathbb{R}^{\|C(u_i)^+\|\times\|A(u_i,C(u_i)^+)\|}$.
$\mathbf{E^A}(u_i, C(u_i)^+)$	It represents u_i's action engagement in all enrolled courses, where $\mathbf{E^A}(u_i, C(u_i)^+) = [e_{i1}^a, e_{i2}^a, \ldots, e_{i\|C(u_i)^+\|}^a]$.
$\mathbf{E^T}(u_i, C(u_i)^+)$	The time engagement of u_i in all enrolled courses.
$\mathbf{X}_i^{\mathbf{A}}$	The engagement of u_i on all courses, where $\mathbf{X}_i^{\mathbf{A}} = [x_{i1}, x_{i2}, \ldots, x_{ij}, \ldots, x_{i\|C\|}]$, where $0 \le x_{ij} \le 1$. The closer x_{ij} is to 1, the higher u_i's engagement in c_j.
$\widehat{\mathbf{X}^{\mathbf{A}}}_i$	The engagement of u_i on all courses predicted by the ENN model, where $\widehat{\mathbf{X}^{\mathbf{A}}}_i = [\hat{x}_{i1}, \hat{x}_{i2}, \ldots, \hat{x}_{ij}, \ldots, \hat{x}_{i\|C\|}]$.
PCR@$K\left(u_i\right)$	The set of making personalized courses recommended by prediction of engagement to u_i.

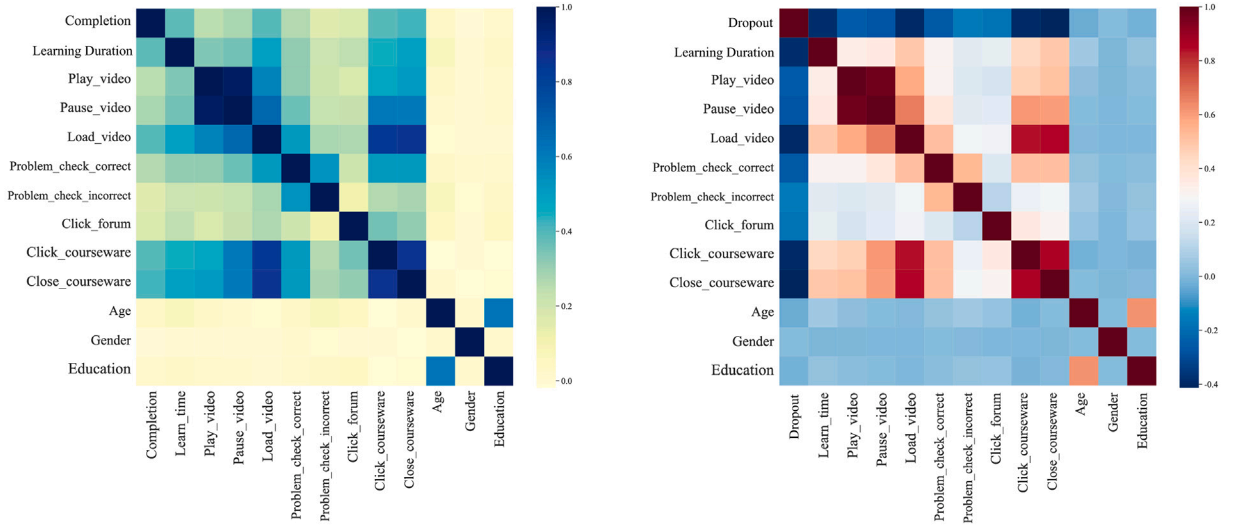
RQ 2. How to predict u_i's engagement in unenrolled courses? The task can be described as: Given the engagement in enrolled courses $\mathbf{X}_i^{\mathbf{A}+}$ by QE, the incidence matrix of learners and courses **UC**, and average aggregation information matrix **UA**, **CA**, how to design the ENN model to predict u_i's engagement in unenrolled courses?

RQ 3. How to make personalized courses recommended by prediction of engagement to u_i, such that u_i can complete the recommended courses as much as possible? Given the predicted u_i's engagement in unenrolled courses based on ENN, the purpose is how to achieve personalized recommendation for u_i ($u_i \in U$) so that u_i completes the recommended courses as much as possible, thus reducing the dropout rate of the MOOC platforms?

To address these three research questions, this paper use the QE method to quantify the learner's engagement in enrolled courses in Section 4; Then use the ENN model to predict the learner's engagement in unenrolled courses in Section 5; Finally, based on the magnitude of engagement predicted in Section 6, personalized course recommendations for learners to reduce dropout rates. The overall model architecture of this paper is shown in Fig. 3, which also shows the relationship between the three sub-problems. The main symbols and descriptions involved in this paper are show in the Table 2.

4. Quantitation of engagement in enrolled courses

This section details methods for quantifying learner engagement in enrolled courses and finds a relationship between quantified engagement and dropout rates.



(a) Completion and Characteristics

(b) Dropout and Characteristics

Fig. 4. (a) and (b) are the relationships between learners' attribute characteristics, action characteristics, learning duration and completion rates, and dropout rates on XuetangX dataset, respectively.

4.1. Datasets

The analysis of this work analyzed two datasets, Dropout Predication Dataset [9] in XuetangX and KDDCUP Data[10] mentioned in the Feng et al. (2019). They will be referred to as XuetangX and KDDCUP, respectively, in the paper, and will be introduced in the following:

XuetangX: It offers more than 1,000 courses and attracts more than 10 million enrolled users, each user can enroll in one or more courses. When one studying a course, the system records multiple types of behavior characteristics, and each type of behavior contains multiple actions: videos ($Play\_video$, $Stop\_video$, $Pause\_video$, $Seek\_video$, and $Load\_video$), Problem ($Problem\_check$, $Problem\_save$, $Problem\_check\_correct$, and $Problem\_check\_incorrect$), Forum ($Click\_forum$ and $Close\_forum$), and Courseware ($Click\_courseware$ and $Close\_courseware$). The dataset also contains learner dropouts, where 1 means dropout and 0 means no dropout. Also included are the learner's start time for each action and the end time. This dataset contains 60884 learners and 247 courses.

KDDCUP: It provides information on the learning behavior of 39 courses in the XuetangX half-year, which was used for KDDCUP 2015. The dataset includes five behaviors: "Video", "Problem", "Wiki", "Forum", and "Discussion". Also included in this dataset is whether or not the learner dropouts.

In this paper, considering that learners' engagement in one part of courses is required to predict another part of courses, learners who choose less than 2 courses are excluded. In the end, 247 courses and 28989 learners are selected in the XuetangX dataset, in the KDDCUP dataset choose 39 courses and 18688 learners. Then these data are divided into Training and Test sets in the ratio of 8:2.

4.2. Feature analysis

In traditional classes, the number of times a student raises his hand to answer questions can reflect whether he is active in class. Identifying these cues in MOOCs is challenging, but the large amount of available data can offset the loss of face-to-face communication, learners interact with the class by clicking and operating the MOOC platforms, such as "Play_video", "Click_forum", "Click_courseware", and so on. Therefore, clicking and operating can reflect their enthusiasm for an online class. At the same time, the learning duration of a certain course can also directly reflect whether learners are willing to spend time on the course.

This paper will analyze the relationship between learners' attribute characteristics, action characteristics, learning duration, completion rate, and dropout rate in the XuetangX dataset mentioned in Section 4.1, which includes both Training and Test sets, respectively. It can be seen from Fig. 4(a) that learners' action characteristics and learning duration have a strong correlation with completion rates, while learners' attribute characteristics have a relatively weak relationship with completion rates. Fig. 4(b) illustrates that there is also a strong correlation between the above characteristics and dropout rates. Therefore, this paper chooses

9 http://moocdata.cn/data/user-activity/DropoutPredictionDataset.
10 http://moocdata.cn/data/user-activity/KDDCUPData.

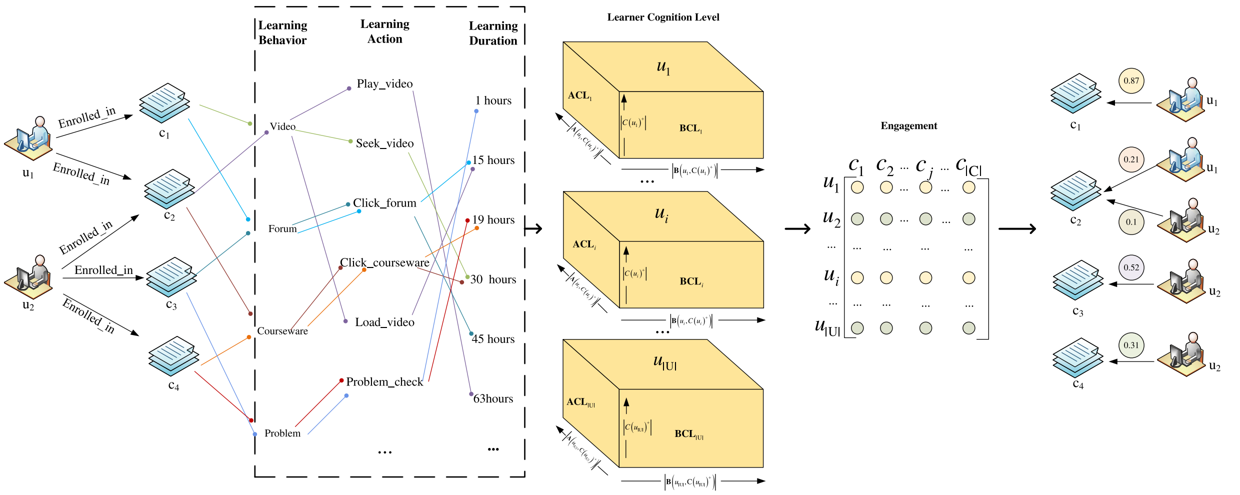


Fig. 5. The framework of QE calculation process.

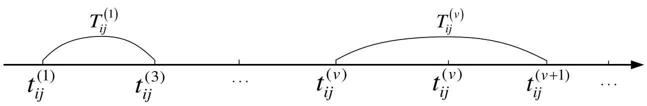


Fig. 6. Schematic diagram of learning duration calculation.

to use learning action data and learning duration to quantify learners' engagement, making the quantified engagement more related to the completion and dropout rates.

4.3. Steps to quantify engagement

This paper quantifies the learners' action data on the course and the learning duration they visit the course as a score between 0 and 1, which is called engagement, to measure whether they are willing to invest time on the course. It is also a reflection of learners' satisfaction with the courses they have learned. The specific quantitative method is shown in Fig. 5, and the steps are as follows,

Step 1. Count the learning action type clicks of u_i in the enrolled course. When u_i enrolled in c_j, there will be multiple actions. Firstly, the click times of each action need to be counted, which is expressed as $N\left(u_i, A\left(u_i, c_j\right)\right) = \{n_{ij}^{(1)}, n_{ij}^{(2)}, \ldots, n_{ij}^{(p)}, \ldots, n_{ij}^{(|A(u_i,c_j)|)}\}$, where $n_{ij}^{(p)}$ represents the learning action type clicks of the pth action of u_i in c_j.

Step 2. Calculate the learning duration u_i spends on a course. This paper records the difference between two consecutive actions timestamps of u_i on c_j, denoted as $t_{ij}^{(p+1)} - t_{ij}^{(p)}$, where $t_{ij}^{(p)}$ and $t_{ij}^{(p+1)}$ represents the start time of the p and $p+1$ action types of the u_i on c_j, respectively. Finally, this paper adds the difference between all the timestamps on c_j to calculate the learning duration of u_i. The formula is as follows:

$$t_{ij} = \sum_{p=1}^{|A(u_i,c_j)|} t_{ij}^{(p+1)} - t_{ij}^{(p)}, \tag{1}$$

$$T\left(u_i, C(u_i)^+\right) = \left\{t_{ij} | u_i \text{ enrolled in } c_j, c_j \in C(u_i)^+\right\}, \tag{2}$$

where $t_{ij}^{(p)}$ and $t_{ij}^{(p+1)}$ represents the start time of the p and $p+1$ action types of the u_i on c_j, respectively. The p action type and $p+1$ action type are continuous in time. t_{ij} represents the total learning duration of u_i on c_j. $T\left(u_i, C(u_i)^+\right)$ denotes the set of learning duration for u_i in enrolled courses. The specific calculation diagram is shown in Fig. 6.

Step 3. Calculate the weight matrix of learning action type about u_i. Learners' actions in different courses are different and there is a gap in the number of clicks. Different actions have different influences on quantified u_i's engagement. Therefore, it is necessary to further calculate the weight of each learning action in all actions in this course, and then form the weight matrix of learning action $\mathbf{Z^A}_i = (z_{jp}^{(i)})_{|C(u_i)^+| \times |A(u_i, C(u_i)^+)|}$, $z_{jp}^{(i)} = n_{ij}^{(p)} / \sum_{p=1}^{|A(u_i,c_j)|} n_{ij}^{(p)}$, $z_{jp}^{(i)}$ represents the weight value of the pth action of u_i on c_j, where $\sum_{p=1}^{|A(u_i,c_j)|} z_{jp}^{(i)} = 1$.

Step 4. Calculate the activity matrix of learning action type about u_i. Learners have different learning habits, for the same course, more active learners tend to make more actions. Therefore, this paper analyzes each action to judge the activity of u_i

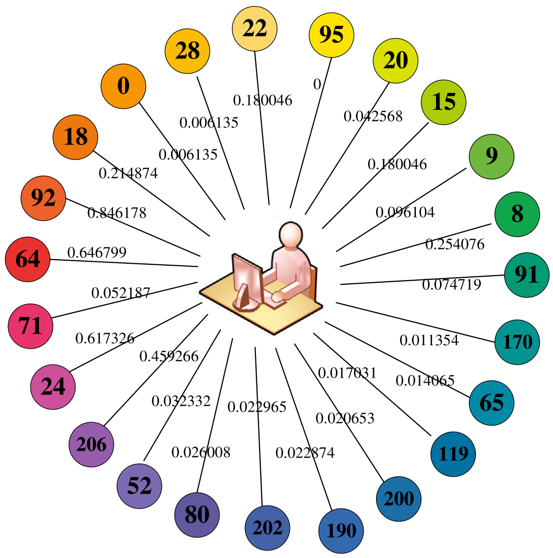


Fig. 7. Schematic diagram of engagement. The weight on each line reflects the engagement to join the corresponding courses, and the circles represent distinct courses.

about action in a certain course. The activity matrix of u_i in all enrolled courses is denoted by $\mathbf{M^A}_i = (m_{jp}^{(i)})_{|C(u_i)^+|\times|A(u_i,C(u_i)^+)|}$, $m_{jp}^{(i)} = n_{ij}^{(p)}/max_i\left\{n_{ij}^{(p)}, u_i \in U(c_j)^+\right\}$, while $max_i\left\{n_{ij}^{(p)}, u_i \in U(c_j)^+\right\}$ represents the maximum click number of the pth learning action for all learners who enrolled in c_j.

Step 5. Calculate the learning action cognition level about u_i. From the above, it can calculate the weight matrix $\mathbf{Z^A}_i$ and the activity matrix $\mathbf{M^A}_i$ of each learner's learning action. The following Eq. (3) can be used to calculate u_i's learning action cognition level in all enrolled courses.

$$\mathbf{ACL}_i = \mathbf{Z^A}_i \odot \mathbf{M^A}_i, \tag{3}$$

where $\mathbf{ACL}_i = (acl_{jp}^{(i)})_{|C(u_i)^+|\times|A(u_i,C(u_i)^+)|}$, $acl_{jp}^{(i)}$ is the result obtained by multiplying the corresponding positions of the $\mathbf{Z^A}_i$ and $\mathbf{M^A}_i$, it represents the learning action cognition level of p-th action about u_i on c_j. $\odot$ is the Hadamard product of them, each row of the $\mathbf{ACL}_i$ can be represented as $\mathbf{ACL}_{ij} = [acl_{j1}^{(i)}, acl_{j2}^{(i)}, \dots, acl_{jp}^{(i)}, \dots, acl_{j|A(u_i,c_j)|}^{(i)}]$.

Step 6. Calculate u_i's action engagement using learning action cognition level. $\mathbf{E^A}(u_i, C(u_i)^+)$ represents u_i's action engagement in all enrolled courses. The calculating process using learning action cognition level is given as follows by Eq. (4).

$$\mathbf{E^A}(u_i, C(u_i)^+) = \left\{ \sum_{p=1}^{|A(u_i,c_j)|} acl_{jp}^{(i)} \,\middle|\, u_i \text{ enrolled in } c_j,\ c_j \in C(u_i)^+ \right\}, \tag{4}$$

where $\mathbf{E^A}(u_i, C(u_i)^+) = [e_{i1}^a, e_{i2}^a, \dots, e_{i|C(u_i)^+|}^a]$.

Step 7. Calculate u_i's time engagement using learning duration data. Assuming a total of $|U|$ learners are studying in $|C|$ courses, there are $|U| * |C|$ records. This paper takes the top 30% of active learning records and find the average as a criterion for the degree of activity. The time engagement is then calculated using the following formula,

$$e_{ij}^t = \frac{t_{ij}}{Mean(\text{top}-T(|U| * |C|) * 30\%)}, t_{ij} \in T(u_i, C(u_i)^+), \tag{5}$$

where t_{ij} represents the learning duration of u_i in c_j, $Mean(\text{top}-T_{(|U|*|C|*30\%)})$ represents the average learning duration of the top 30% of active records, and time engagement can be denoted as $\mathbf{E^T}(u_i, C(u_i)^+) = \{e_{ij}^t | u_i \text{ enrolled in } c_j, c_j \in C(u_i)^+\}$.

Step 8. Calculate the engagement. In the previous step, this paper calculated the action and time engagement respectively. This paper added the two in different proportions to calculate the engagement of u_i on c_j. The formula is as follows:

$$x_{ij} = e_{ij}^a * \mu + e_{ij}^t(1-\mu), \tag{6}$$

when $\mu = 60\%$, the subsequent recommendation results are better. $\mathbf{X^A}_i^+ = [x_{i1}, x_{i2}, \dots, x_{ij}, \dots, x_{i|C(u_i)^+|}]$ represents the u_i's engagement in all enrolled courses.

A real-world case in the XuetangX is shown in Fig. 7, where the engagement of a learner in various courses is quantified. The weight on each line reflects the engagement to join the corresponding courses, and the circles represent distinct courses. According

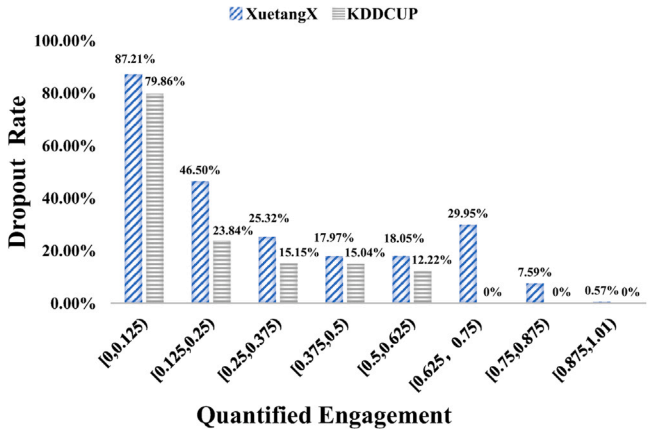


Fig. 8. The relationship between quantified engagement by the ENN model and dropout rates on the XuetangX and KDDCUP datasets. The x-axis is the quantified interval of engagement in enrolled courses, y-axis is the dropout rates at each interval.

to the quantified engagement, it is easy to obtain the learning status of this learner in different courses he enrolled in and then observed whether they are willing to spend time on the courses they have learned.

4.4. Relationship between quantified engagement and dropout rates

Engagement can be a good measure of whether a learner is willing to spend enough time on a particular course, and it also means whether the learner has the opportunity to complete the course enrolled. This paper quantified learners' learning action and learning duration data as engagement by analyzing the dataset mentioned in Section 4.1. As can be seen in Fig. 8, the quantified engagement is closely related to the dropout rate.

Based on the experimental results in Fig. 8, this paper has the following observation.

Observation 1. *The dropout rates of learners in* ***enrolled courses*** *reduce gradually with the increase in their* ***quantified engagement****.*

5. Prediction of engagement in unenrolled courses using neural network

There are no actions recorded when learners are not enrolled in the class, so they cannot be quantified by QE. This section proposes a model of ENN, aiming at predicting the engagement of learners in unenrolled courses.

5.1. Overview of framework

The general structure of Fig. 9 for the ENN model, it can predict the engagement of learners in unenrolled courses. $\mathbf{X^A}$ represents the engagement matrix of all learners in all courses. Gray indicates that the learner is enrolled in the course, and white represents unenrolled in the course. Assuming that there are $|C|$ courses in the MOOC platforms, $C = \{c_1, c_2, c_3, \dots, c_{|C|}\}$, different learners choose different courses. Here u_i is taken as an example. u_i chooses 5 of them $C_i = \{c_1, c_3, c_4, c_5, c_6\}$, engagement can be calculated for each course. Firstly, take $\mathbf{X}_i^{\mathbf{A}+}$ out of the $\mathbf{X^A}$ matrix, because the ENN model needs to predict engagement in one part of the class from engagement in another part of the class, $\mathbf{X}_i^{\mathbf{A}+}$ needs to be divided into two parts $\mathbf{C}_{X(i)}$ and $\mathbf{C}_{Y(i)}$. The divided method can use α-partition method, which is similar to the cross-validation partitioning of data. As shown in Fig. 9, $\alpha = 3$, so that one valid data can be trained three times, and the model can obtain the optimal result after training. The result of the first division is $\mathbf{C}_{X(i)} = \{c_3, c_4, c_6\}$, $\mathbf{C}_{Y(i)} = \{c_1, c_5\}$. The second division after the results as follows $\mathbf{C}_{X(i)} = \{c_1, c_3, c_5, \}$, $\mathbf{C}_{Y(i)} = \{c_4, c_6\}$. The result of the third division after $\mathbf{C}_{X(i)} = \{c_1, c_4, c_5, c_6\}$, $\mathbf{C}_{Y(i)} = \{c_3\}$, where each divided course selection record is independent in the subsequent processing. As we can see, each course can be divided into $\mathbf{C}_{Y(i)}$. As different learners choose different courses, the courses chosen by all learners will cover all courses after model training. Thus, the α-partition method improves the predictive power of ENN.

5.2. Neural network structure

Firstly, the average aggregation information of each learner's corresponding courses in $\mathbf{C}_{X(i)}$ is extracted from the matrix $\mathbf{CA}$, expressed as $\mathbf{CA}_i = (ca_{jp}^{(i)})_{|C(u_i)^+|\times|A(u_i,C(u_i)^+)|}$, where the course in $|C(u_i)^+|$ is consistent with the course in $\mathbf{C}_{X(i)}$. Moreover, the softmax processing of $\mathbf{CA}_i$ is denoted as $\mathbf{CA}_s = (ca_{jp}^{(s_i)})_{|\mathbf{C}_{X(i)}|\times|A|}$, and the formula is as follows:

$$ca_{jp}^{(s_i)} = \frac{\exp(ca_{jp}^{(i)})}{\sum_{p=1}^{|A|} \exp\left(ca_{jp}^{(i)}\right)}. \tag{7}$$

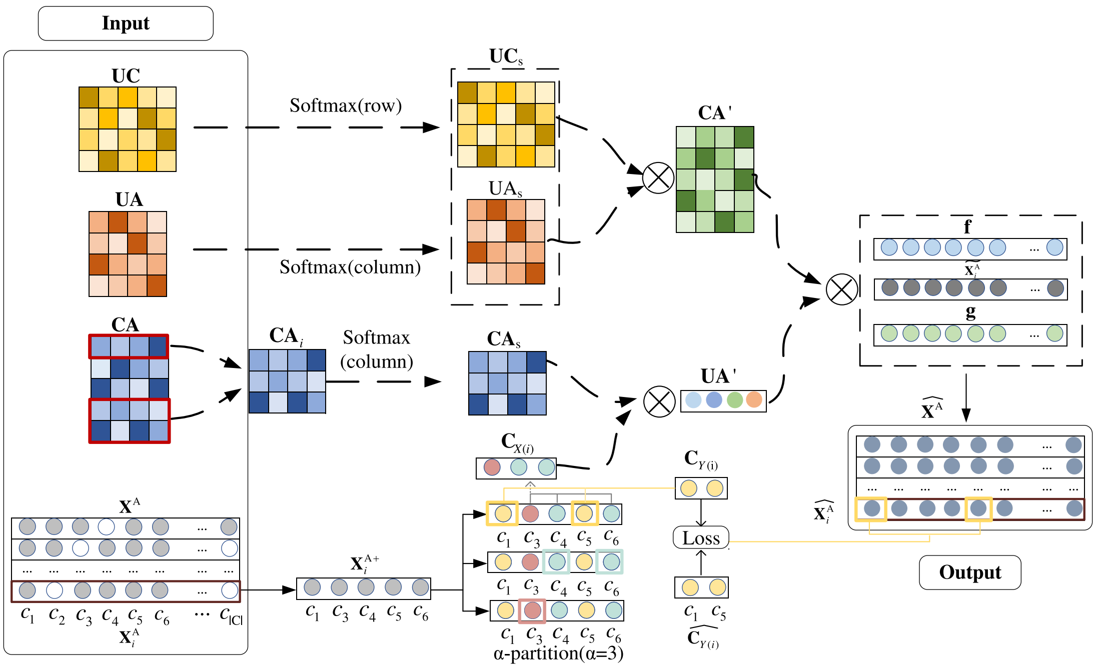


Fig. 9. The model of ENN. ● is the course for which the learner has enrolled. ○ is not enrolled.

UA's softmax processing is similar as Eq. (7), it is carried out for each column to make its value distributed between $\{0, 1\}$. The result is denoted as $\mathbf{UA}_s = (ua_{ip}^{(s)})_{|U|\times|A|} \in \mathbb{R}^{|U|\times|A|}$. **UC** is denoted as $\mathbf{UC}_s = (uc_{ij}^{(s)})_{|U|\times|C|}$ after softmax, and the formula is shown below, where $uc_{ij}^{(s)} \in \{0, 1\}$.

$$\mathbf{UC}_s = softmax_{(axis=0)}(\mathbf{UC}). \tag{8}$$

Next, obtain the degree of participation of u_i in various learning actions $\mathbf{UA}' = \mathbb{R}^{1\times|A|}$ in the learned lessons through the $\mathbf{CA}_s$ and $\mathbf{C}_{X(i)}$. It shows what actions learners are more inclined to do on the learned course, which is very useful for us to analyze what actions learners will do in the unenrolled course. As shown in the Eq. (9),

$$\mathbf{UA}' = \mathbf{C}_{X(i)} \otimes \mathbf{CA}_s, \tag{9}$$

where $\otimes$ is matrix multiplication. Through $\mathbf{UA}_s$ and $\mathbf{UC}_s$ get the degree of participation of all learners in each action in all courses, which is defined as $\mathbf{CA}' = \left(ca'_{jp}\right)_{|C|\times|A|}$. And then analyze which actions most learners are more inclined to, to infer the possible actions of u_i. The formula is shown in Eq. (10), where T is the transpose of $\mathbf{UC}_s$.

$$\mathbf{CA}' = \mathbf{UC}_s{}^{\mathrm{T}} \otimes \mathbf{UA}_s. \tag{10}$$

Then, infer the engagement of the u_i in all courses through the degree of the u_i's tendency of action in the learned courses and other learners in all courses. The formula is shown below,

$$\widetilde{\mathbf{X}^{\mathbf{A}}}_i = \mathbf{UA}' \otimes \mathbf{CA}'^{\mathrm{T}}. \tag{11}$$

Considering that learners will be affected by non-learning factors in the process of course learning, this paper introduces two offset vector $\mathbf{f}' \in \mathbb{R}^{|C|}$, $\mathbf{g}' \in \mathbb{R}^{|C|}$, each element in $\mathbf{f}'$ and $\mathbf{g}'$ has a slightly smaller initial value, such as $f_j{}' = -2\,(1 \le j \le |C|)$ and $g_j{}' = -2\,(1 \le j \le |C|)$. $\mathbf{f}'$ considers some interference factors that learners like a certain course but have a low engagement in it. For example, learners who are sick or suffer from epidemics cannot finish the course on time. $\mathbf{g}'$ considers uncontrollable factors that learners have a high engagement in a course even if they do not like it, such as the course is related to offline final exams.

After adding offsets $\mathbf{f}'$ and $\mathbf{g}'$, need to sigmoid them first, $\mathbf{f} = sigmoid(\mathbf{f}')$, $\mathbf{g} = sigmoid(\mathbf{g}')$. Where $\mathbf{f}' \in \mathbb{R}^{|C|}$ is the parameter of the model, which is used to calculate $\mathbf{f}$, formula such as:

$$f_j = \frac{1}{1 + \exp(-f_j{}')}, \tag{12}$$

$\mathbf{g}' \in \mathbb{R}^{|C|}$ is the parameter of the model, which is used to calculate $\mathbf{g}$, same as Eq. (12).

Through the pre-estimation of offset, the final output result of our model is

$$\widehat{\mathbf{X}^{\mathbf{A}}}_i = \widetilde{\mathbf{X}^{\mathbf{A}}}_i \times (1 - \mathbf{f}) + (1 - \widetilde{\mathbf{X}^{\mathbf{A}}}_i) \times \mathbf{g}, \tag{13}$$

where $\times$ represents the multiplication of elements at the corresponding position of the vector, 1 represents a vector of all ones.

Algorithm 1 The Prediction Framework of ENN.

Input: ${\mathbf{X^A}}_i^+$: The learners' engagement in enrolled course; **CA**: Learner aggregation matrix; **UC**: the incidence matrix about learners and courses; **UA**: Course aggregation matrix; α: Random α-partition divides the data into α parts.

Output: $\widehat{\mathbf{X^A}}_i$: The predicted value of learners' engagement in all course.

1: $f'_j \leftarrow -2, g'_j \leftarrow -2(1 \le j \le |N|)$;

2: $\widehat{\mathbf{X^A}} \leftarrow \emptyset, Loss \leftarrow 0$;

3: **for** each epoch from 1 to max epoch **do**

4: **for** each ${\mathbf{X^A}}_i^+$ in $\mathbf{X^A}$ **do**

5: Divide the matrix of non-negative elements into $\mathbf{C}_{X(i)}$ and $\mathbf{C}_{Y(i)}$ by α-partition, there will produce α $\mathbf{C}_{X(i)}$ and α corresponding $\mathbf{C}_{Y(i)}$, denoted as $\{\mathbf{C}^1_{X(i)}, \mathbf{C}^2_{X(i)}, \cdots, \mathbf{C}^{\alpha}_{X(i)}\}$ and $\{\mathbf{C}^1_{Y(i)}, \mathbf{C}^2_{Y(i)}, \cdots, \mathbf{C}^{\alpha}_{Y(i)}\}$ respectively;

6: $Loss_i \leftarrow 0$;

7: **for** α from 1 to α **do**

8: $\widehat{\mathbf{X^A}}_i \leftarrow \emptyset$;

9: Extract the parameter only related to $\mathbf{C}^{\alpha}_{X(i)}$ from **CA** and denoted as $\mathbf{CA}_i$;

10: $\mathbf{CA}_s \leftarrow softmax(\mathbf{CA}_i, axis = 1)$;

11: $\mathbf{UC}_s \leftarrow softmax(\mathbf{UC}, axis = 0)$;

12: $\mathbf{UA}_s \leftarrow softmax(\mathbf{UA}, axis = 1)$;

13: $\mathbf{UA}' \leftarrow \mathbf{C}^{\alpha}_{X(i)} \otimes \mathbf{CA}_s$;

14: $\mathbf{CA}' \leftarrow {\mathbf{UC}_s}^{\mathrm{T}} \otimes \mathbf{UA}_s$;

15: $\widetilde{\mathbf{X^A}}_i \leftarrow \mathbf{UA}' \otimes \mathbf{CA}'^{T}$;

16: $\mathbf{f} \leftarrow sigmoid(\mathbf{f}')$;

17: $\mathbf{g} \leftarrow sigmoid(\mathbf{g}')$;

18: $\widehat{\mathbf{X^A}}_i = \widetilde{\mathbf{X^A}}_i \times (1 - \mathbf{f}) + (1 - \widetilde{\mathbf{X^A}}_i) \times \mathbf{g}$;

19: Extract the engagement on the matrix $\mathbf{C}^{\alpha}_{Y(i)}$ from $\widehat{\mathbf{X^A}}$ and denoted as $\widehat{\mathbf{X^A}}_i$;

20: $Loss_i \leftarrow Loss_i - \frac{1}{|\mathbf{C}_{Y(i)}|} \sum_{\substack{x_{ij} \in \mathbf{C}_{Y(i)} \\ \hat{x}_{ij} \in \widehat{\mathbf{C}_{Y(i)}}}} \left[x_{ij} \log\left(\hat{x}_{ij}\right) + \left(1 - x_{ij}\right) \log(1 - \hat{x}_{ij})\right]$;

21: **end for**

22: $Loss \leftarrow Loss + \frac{Loss_i}{\alpha}$;

23: **if** epoch = max epoch **then**

24: Add $\widehat{\mathbf{X^A}}_i$ to matrix $\widehat{\mathbf{X}^A}$;

25: **end if**

26: **end for**

27: $Loss \leftarrow \frac{Loss}{|U|}$;

28: Update $\mathbf{f}', \mathbf{g}'$ and $Loss$;

29: **end for**

30: return $\widehat{\mathbf{X}^A}$.

Finally, when calculating the loss of the ENN, extracted the engagement on the matrix $\mathbf{C}_{Y(i)}$ from $\widehat{\mathbf{X^A}}$ and denoted as $\widehat{\mathbf{X^A}}_i$. As shown in Fig. 9, $\widehat{\mathbf{X^A}}_i = [\hat{x}_{i1}, \hat{x}_{i2}, \ldots, \hat{x}_{i|C|}]$, $\widehat{C_{Y(i)}} = \left\{\hat{x}_{i1}, \hat{x}_{i5}\right\}$, $C_{Y(i)} = \left\{x_{i1}, x_{i5}\right\}$, by comparing the difference between $\widehat{C_{Y(i)}}$ and $C_{Y(i)}$, the loss of u_i on each course is calculated, as shown in Eq. (14),

$$Loss_i = -\frac{1}{\alpha|C_{Y(i)}|} \sum_{\substack{x_{ij} \in C_{Y(i)} \\ \hat{x}_{ij} \in \widehat{C_{Y(i)}}}} \left[x_{ij} \log\left(\hat{x}_{ij}\right) + \left(1 - x_{ij}\right) \log(1 - \hat{x}_{ij})\right]. \tag{14}$$

The loss of the ENN on all courses' engagement for all learner is

$$Loss = \frac{1}{|U|} \sum_{u_i \in U} Loss_i. \tag{15}$$

The optimization goal of the model is to minimize $Loss$. The above process is the forward propagation of the model, in the process of back propagation, the gradient of $Loss$ to model parameters $\mathbf{f}'$ and $\mathbf{g}'$ is calculated successively, and the model parameters are updated by the Adam algorithm. The specific flow chart of the training algorithm is shown in Algorithm 1. Lines 1–2 initialize the model parameters; Lines 4–5 divide each learner's data into two parts by the α-partition method; Line 6 initializes $Loss_i$; Lines 7–20 describe the processing of the ENN model to predict the engagement of learners in unenrolled courses; Lines 23–24 are to obtain the engagement of learners in each course; Lines 27–28 are back propagates and updates the parameters.

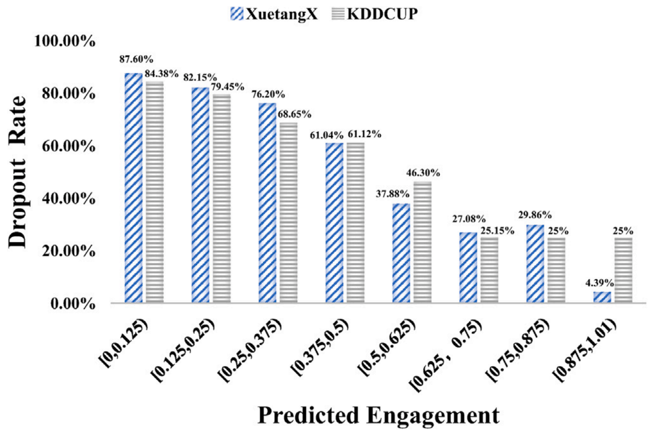


Fig. 10. The relationship between predicted engagement by the ENN model and dropout rates on the XuetangX and KDDCUP datasets. The x-axis is the predicted interval of engagement in unenrolled courses, the y-axis is the dropout rates at each interval.

5.3. Relationship between predicted engagement and dropout rates

Through extensive experiments, this paper obtained Fig. 10. These data come from learners and courses in the XuetangX dataset mentioned in Section 4.1, where the data from the Test set is used here, and it can be seen that the overall trend of the relationship between predicted engagement and dropout rate is consistent with Observation 1, and thus has the following observation.

Observation 2. *The dropout rates of learners in* ***unenrolled courses*** *reduce gradually with the increase in their* ***predicted engagement***.

6. Applied to personalized course recommendation by prediction of engagement to reduce dropout in MOOCs

Engagement can be a good measure of whether a learner is willing to spend enough time on a particular course, and it also means whether the learner has the opportunity to complete the course enrolled. Based on Observation 2, the dropout rates of learners in unenrolled courses reduce gradually with the increase in their predicted engagement. The experimental proof is shown in Fig. 10. Hence, this paper proposes a personalized course recommendation framework, which uses predicted engagement to personalize course recommendations so that learners are more likely to complete the course. This reduces the dropout rate of the MOOC platforms. The stages are as follows:

Stage 1. Using the QE method to quantify the engagement of u_i in enrolled courses. $\mathrm{QE}(N\left(u_i, A\left(u_i, C(u_i)^{+}\right)\right), T\left(u_i, C(u_i)^{+}\right))$ represents quantify the engagement of u_i in enrolled courses, This paper describes the quantification process as follows:

Iuput: $N\left(u_i, A\left(u_i, C(u_i)^{+}\right)\right)$, the number of learning action type clicks about u_i in his enrolled course. $T\left(u_i, C(u_i)^{+}\right)$, the learning duration for different courses.

$$\mathbf{X^{A}}_i^{+} = \mathrm{QE}\left(N\left(u_i, A\left(u_i, C(u_i)^{+}\right)\right), T\left(u_i, C(u_i)^{+}\right)\right). \tag{16}$$

Output: $\mathbf{X^{A}}_i^{+} = [x_{i1}, x_{i2}, \ldots, x_{ij}, \ldots, x_{i|C(u_i)^{+}|}]$, the u_i's engagement in each enrolled course quantified by QE method.

Stage 2. Using the ENN model to predict the engagement of u_i in unenrolled courses. In this paper, the output of QE is taken as one of the inputs of ENN. $\mathrm{ENN}\left(\mathbf{UC}, \mathbf{UA}, \mathbf{CA}, \mathbf{X^{A}}_i^{+}\right)$ indicates the prediction of u_i's engagement in unenrolled courses by **UC**, **UA**, **CA** and $\mathbf{X^{A}}_i{}^{+}$. The solution process of ENN can be expressed as the following function:

Input: $\mathbf{X^{A}}_i{}^{+}$, u_i's engagement in enrolled courses. **UC**, the incidence matrix about learners and courses, **UA** and **CA** are the average aggregation information matrix.

$$\widehat{\mathbf{X^{A}}_i^{-}} = \mathrm{ENN}\left(\mathbf{UC}, \mathbf{UA}, \mathbf{CA}, \mathbf{X^{A}}_i^{+}\right). \tag{17}$$

Output: The engagement of u_i in all courses, $\widehat{\mathbf{X^{A}}_i} = [\hat{x}_{i1}, \hat{x}_{i2}, \ldots, \hat{x}_{ij}, \ldots, \hat{x}_{i|C|}]$, where $\widehat{\mathbf{X^{A}}_i^{-}} \in \widehat{\mathbf{X^{A}}_i}$.

Stage 3. Using the engagement of u_i in unenrolled courses predicted by ENN to make personalized course recommendations to reduce dropout rates. According to Observation 2, this stage recommends top$-K$ courses with high predicted engagement in unenrolled courses to u_i. Here, top$-K(\widehat{\mathbf{X^{A}}_i^{-}})$ denotes top$-K$ courses with high predicted engagement in unenrolled courses for u_i. Thus, this paper has the following equation Eq. (18):

Input: $\widehat{\mathbf{X^{A}}_i^{-}}$, the engagement of u_i in unenrolled courses.

$$\mathrm{PCR@}K\left(u_i\right) = \mathrm{top}{-}K(\widehat{\mathbf{X^{A}}_i^{-}}). \tag{18}$$

Output: $\mathrm{PCR@}K\left(u_i\right)$, the set of making personalized courses recommended by prediction of engagement to u_i.

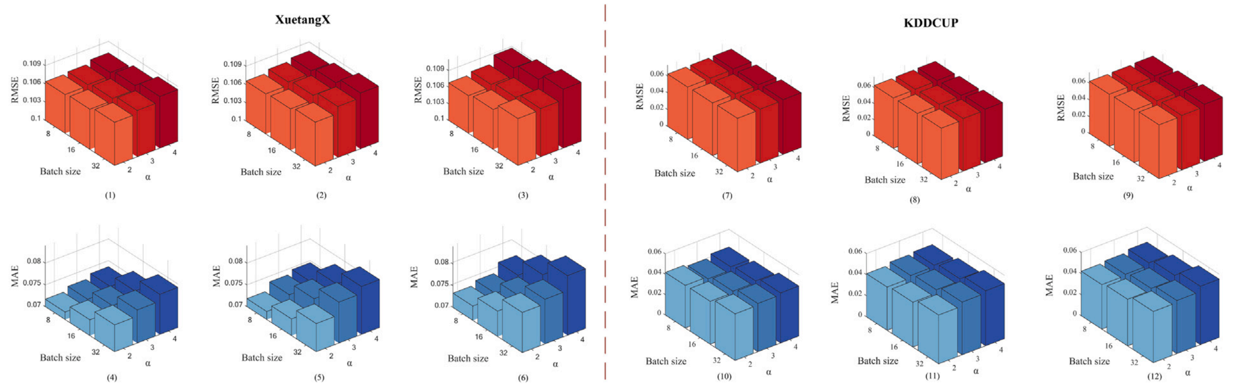


Fig. 11. The variation of RMSE and MAE for different values of hyperparameters Batch size, α, and Train/Test set ratio respectively on the XuetangX and KDDCUP datasets.

7. Experiments

In this section, the datasets and splits, evaluation metrics, baselines, experimental settings, results analysis, and quantitative relationship interpretability experiments are described.

7.1. Datasets and experimental setup

Datasets introduction: The analysis of this work was performed on XuetangX and KDDCUP dataset, the specific description of the dataset is given in Section 4.1.

Datasets splits: This paper divided the whole data set D into Training set I and Test set S respectively according to the ratio of $\chi : \beta$ for the course chosen by each learner. If the learner u_i is registered for 8 courses, the Training set contains $8 * \chi$ courses and the Test set contains $8 * \beta$ courses, where $8 * \chi + 8 * \beta = 8$.

Experimental configuration: The model in this paper was implemented using Python 3.8.12 and Pytorch 1.10.0. All features were standardized before entering QE and ENN, and Adam was used to optimizing the model. The model is built based on the server Intel Core I7-7820x CPU @ 3.60GHz×16, 64 GB Memory, NVIDIA GeForce RTX2080, Windows10 64-bit.

7.2. Evaluation for engagement prediction

7.2.1. Evaluation metrics for engagement prediction

In this paper, the ENN model has a good predictive function. The input and output of the model are the learners' engagement in a certain course, which is recorded as a score between 0–1, indicating the active degree of learners in learning the course, it also reflects the willingness of learners to spend time in enrolled courses. Therefore, the paper first chooses MAE and RMSE to measure the numerical difference between learners' predicted engagement and their actual engagement. The Eq. (19) is as follows:

$$MAE = \frac{\sum_{i=1}^{|U|}\sum_{j=1}^{|C|} |\hat{x}_{ij} - x_{ij}|}{|U| \times |C|}, \qquad RMSE = \sqrt{\frac{\sum_{i=1}^{|U|}\sum_{j=1}^{|C|} (\hat{x}_{ij} - x_{ij})^2}{|U| \times |C|}}, \tag{19}$$

where $\hat{x}_{ij}$ is the predicted value of learner u_i's engagement on c_j course output by the ENN model, and x_{ij} represent the real value of engagement.

7.2.2. Parameter sensitivity analysis for engagement prediction

The hyperparameters in this paper include Batch size, α, and Train/Test set ratio, where α is the parameter in the random partition. This paper will analyze the performance of the model of ENN by analyzing these hyperparameters, to optimize the model and improve its robustness of the model.

On the XuetangX dataset, as shown in Fig. 11(1)(4), when Train: Test= 8:2, change the values of Batch size and α, the fluctuations of RMSE and MAE are respectively around 0.002 and 0.003; Fig. 11(2)(5) shows that when Train: Test = 7:3, the values of Batch size and α are changed respectively, and the values of RMSE and MAE changed little; Fig. 11(3)(6) shows the effect of changing Batch size and α on RMSE and MAE when Train: Test = 6:4. On the KDDCUP dataset, as shown in Fig. 11(7)(10), when Train: Test= 8:2, change the values of Batch size and α; Fig. 11(8)(11) shows that when Train: Test = 7:3, the values of Batch size and α are changed respectively, and the results hardly change in RMSE and MAE; Fig. 11(9)(12) shows the effect of changing Batch size and α on RMSE and MAE when Train: Test = 6:4, the values of RMSE and MAE changed little.

In conclusion, it is shown on both datasets XuetangX and KDDCUP that Batch size, α, and Train/Test set ratio have little influence on model performance, ENN model is insensitive to hyperparameters.

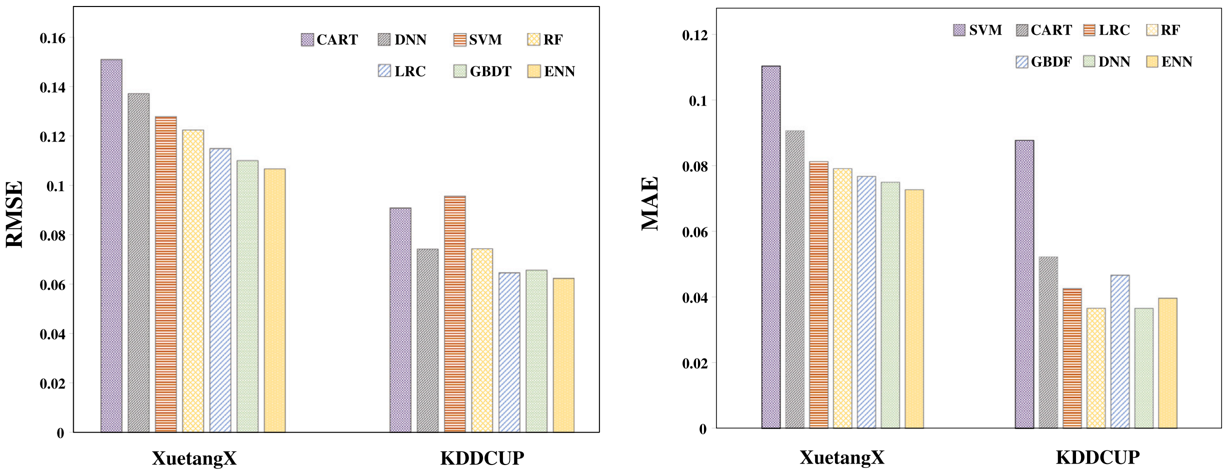


Fig. 12. ENN model predictive performances versus baselines on the XuetangX and KDDCUP datasets. The x-axis represents the different baseline models, the y-axis is the accuracy of RMSE and MAE for different models, respectively.

7.2.3. Baselines for engagement prediction

For the prediction accuracy of engagement, this paper conducted a comparative experiment on the following methods.

- **LR:** Logistic Regression model.
- **SVM:** The support vector machine with linear kernel.
- **RF:** Random Forest model.
- **GBDT:** Gradient Boosting Decision Tree.
- **DNN:** 3-layer deep neural network.
- **CART:** Decision TreeRegressor.

7.2.4. Experimental results for engagement prediction

Fig. 12 shows the results of the ENN model proposed in this paper compared with the baseline models (LR, SVM, RF, GBDT, DNN, CART) on two datasets, where the RMSE and MAE of ENN on the XuetangX dataset are 0.1066 and 0.0727, respectively. On the KDDCUP dataset, the RMSE is 0.0624, the MAE is 0.0396. It can be seen that the ENN model has better predictive performance and outperforms the baseline in predicting the RMSE and MAE of learners' engagement in unenrolled courses on the XuetangX dataset. On the KDDCUP dataset, the results of MAE are slightly worse than DNN and RF, but RMSE outperforms all compared models.

In general, the ENN model has advantages in predicting learner's engagement in unenrolled courses, primarily because it incorporates a broader spectrum of learner-related information into its framework. This includes factors such as the incidence matrix of learners and courses **UC**, the average aggregation matrix of learners **CA**, and the average aggregation matrix of courses **UA**, which enables the model to adapt to the learner's preferences and discern the courses they are more inclined to pursue. Additionally, the model takes into account two non-learning factors $\mathbf{f}'$ and $\mathbf{g}'$ to refine its predictive accuracy.

7.3. Evaluation for personalized course recommendation using engagement

7.3.1. Evaluation metrics for personalized course recommendation using engagement

At the same time, with course recommendations based on the engagement predicted by ENN, this paper will give priority to recommending courses with high engagement, making learners more likely to complete the course. So it can be regarded as a top-K recommendation problem. In this paper, Hit ratio@K, Recall@K, Precision@K, and F1-score@K are selected as evaluation criteria, which are expressed by Eq. (20), Eq. (21), Eq. (22), and Eq. (23) respectively.

$$\text{Hit ratio@}K = \frac{\sum_{u_i \in U} |\text{PCR@}K\left(u_i\right) \cap \left(C\left(u_i\right)^+ - D\left(u_i\right)\right)|}{\sum_{u_i \in U} |C\left(u_i\right)^+|}, \tag{20}$$

where $|\text{PCR@}K\left(u_i\right) \cap \left(C\left(u_i\right)^+ - D\left(u_i\right)\right)|$ represents recommended top-K courses to u_i and the number of courses he completed. $|C\left(u_i\right)^+|$ denotes the number of courses that u_i enrolled in. Hit ratio@K will be denoted by Hit@K.

$$\text{Recall@}K = \frac{1}{|U|} \sum_{u_i \in U} \frac{|\text{PCR@}K\left(u_i\right) \cap \left(C\left(u_i\right)^+ - D\left(u_i\right)\right)|}{|C\left(u_i\right)^+ - D\left(u_i\right)|}, \tag{21}$$

Table 3
When Train: Test = 8:2, Batch size = 8, change the value of α, the recommendation accuracy of different methods was measured by Hit ratio, Recall, Precision, and F1-score using the **XuetangX** dataset. The best results are highlighted in bold.

α	Hit@5	Hit@10	Hit@15	R@5	R@10	R@15	P@5	P@10	P@15	F1@5	F1@10	F1@15
2	16.43%	24.84%	28.74%	17.71%	26.67%	30.76%	16.70%	25.65%	29.59%	17.23%	25.98%	29.97%
3	**21.16%**	**25.77%**	**28.88%**	**22.68%**	**27.61%**	**30.90%**	**21.74%**	**26.53%**	**29.73%**	**22.04%**	**26.88%**	**30.11%**
4	16.88%	24.58%	26.89%	17.90%	26.23%	28.73%	17.02%	25.14%	26.67%	17.30%	25.49%	27.94%

Table 4
When Train: Test = 8:2, Batch size = 8, change the value of α, the recommendation accuracy of different methods was measured by Hit ratio, Recall, Precision, and F1-score using the **KDDCUP** dataset. The best results are highlighted in bold.

α	Hit@3	Hit@5	Hit@10	R@3	R@5	R@10	P@3	P@5	P@10	F1@3	F1@5	F1@10
2	**16.15%**	**28.04%**	**41.17%**	**15.87%**	**27.89%**	**41.18%**	**13.47%**	**23.93%**	**35.21%**	14.22%	**25.19%**	37.11%
3	15.19%	21.60%	34.82%	15.00%	21.27%	34.34%	12.73%	18.06%	29.16%	11.17%	22.30%	**38.48%**
4	14.80%	19.90%	32.30%	14.65%	19.51%	31.64%	12.38%	16.41%	26.51%	**17.39%**	20.69%	37.56%

Table 5
When Batch size = 8, α = 3, change the value of Train/Test set ratio, the recommendation accuracy of different methods was measured by Hit ratio, Recall, Precision, and F1-score using **XuetangX** dataset. The best results are highlighted in bold.

Train:Test	Hit@5	Hit@10	Hit@15	R@5	R@10	R@15	P@5	P@10	P@15	F1@5	F1@10	F1@15
8:2	**21.16%**	**25.77%**	**28.88%**	**22.68%**	**27.61%**	**30.90%**	**21.74%**	**26.53%**	**29.73%**	**22.04%**	**26.88%**	**30.11%**
7:3	18.32%	23.31%	24.93%	19.29%	24.67%	26.36%	15.73%	20.42%	21.82%	16.88%	21.80%	23.29%
6:4	15.96%	25.47%	27.03%	15.67%	25.09%	26.67%	10.72%	17.25%	18.11%	12.31%	19.78%	20.86%

Table 6
When Batch size = 8, α = 2, change the value of the Train/Test set ratio, the recommendation accuracy of different methods was measured by Hit ratio, Recall, Precision, and F1-score using **KDDCUP** dataset. The best results are highlighted in bold.

Train:Test	Hit@3	Hit@5	Hit@10	R@3	R@5	R@10	P@3	P@5	P@10	F1@3	F1@5	F1@10
8:2	**16.15%**	**28.04%**	**41.17%**	**15.87%**	**27.89%**	**41.18%**	**13.47%**	**23.93%**	**35.21%**	**14.22%**	**25.19%**	**37.11%**
7:3	15.84%	22.83%	34.51%	15.68%	22.62%	33.96%	11.21%	15.84%	23.60%	12.58%	17.92%	26.77%
6:4	13.21%	18.86%	30.15%	12.81%	18.16%	28.94%	7.90%	10.49%	16.54%	9.38%	12.82%	20.31%

Table 7
When α = 3, Train : Test = 8:2, change the value of Batch size, The recommendation accuracy of different methods was measured by Hit ratio, Recall, Precision, and F1-score using **XuetangX** dataset. The best results are highlighted in bold.

Batch size	Hit@5	Hit@10	Hit@15	R@5	R@10	R@15	P@5	P@10	P@15	F1@5	F1@10	F1@15
8	**21.16%**	**25.77%**	**28.88%**	**22.68%**	**27.61%**	**30.90%**	**21.74%**	**26.53%**	**29.73%**	**22.04%**	**26.88%**	**30.11%**
16	21.16%	24.08%	28.71%	22.68%	25.73%	30.34%	21.74%	24.32%	29.03%	22.04%	25.01%	30.01%
32	21.16%	24.03%	28.54%	22.68%	25.68%	30.56%	21.74%	24.60%	29.42%	22.04%	24.95%	29.79%

Table 8
When α = 2, Train : Test = 8:2, change the value of Batch size, The recommendation accuracy of different methods was measured by Hit ratio, Recall, Precision, and F1-score using **KDDCUP** dataset. The best results are highlighted in bold.

Batch size	Hit@3	Hit@5	Hit@10	R@3	R@5	R@10	P@3	P@5	P@10	F1@3	F1@5	F1@10
8	**16.15%**	**28.04%**	**41.17%**	**15.87%**	**27.89%**	**41.18%**	**13.47%**	**23.93%**	**35.21%**	**14.22%**	**25.19%**	**37.11%**
16	15.65%	20.96%	32.71%	15.45%	20.70%	32.18%	13.10%	17.58%	27.31%	13.84%	18.57%	28.87%
32	8.82%	17.59%	28.11%	8.49%	17.17%	27.29%	7.27%	14.40%	22.92%	7.64%	15.28%	24.32%

$|C(u_i)^+ - D(u_i)|$ indicates the number of courses completed by u_i, Recall@K denotes the corresponding Recall when recommending K.

$$\text{Precision@}K = \frac{1}{|U|}\sum_{u_i \in U} \frac{|\text{PCR@}K(u_i) \cap \left(C(u_i)^+ - D(u_i)\right)|}{|\text{PCR@}K(u_i)|}, \tag{22}$$

where PCR@$K(u_i)$ is the set of making personalized courses recommended by prediction of engagement to u_i. Precision@K is the precision with which K courses are recommended for learning, which will be denoted by P@K in the future.

$$\text{F1-score@}K = 2 \cdot \frac{\text{Precision@}K \cdot \text{Recall@}K}{\text{Precision@}K + \text{Recall@}K} \tag{23}$$

where F1-score@K is the F1-score with which K courses are recommended for learning, which will be denoted by F1@K in the future.

Table 9
Baselines used to evaluate recommendation systems and the hit standards in existing work.

Paper(Ref)	Hit Standards	Popular	Random	MF	CF	SVD
Jing and Tang (2017)	Recommended courses can only ensure that learners are **interested in** the courses, but cannot guarantee their completion.		✓		✓	
Zhang, Huang, Lv, Liu, and Zhou (2018)					✓	
Boratto, Fenu, and Marras (2019)		✓	✓			✓
Symeonidis, Malakoudis, et al. (2019)				✓		✓
Le, Vo, Nguyen, and Le (2020)				✓		✓
Xu et al. (2021)					✓	
Shao et al. (2021)				✓		
Lin, Feng, et al. (2021)				✓		
Ma et al. (2021)			✓			
Tian and Liu (2021)		✓		✓	✓	
Ban et al. (2022)				✓		
Yang and Cai (2022)					✓	
Lin et al. (2022)				✓		
Wang, Zhu, et al. (2022)				✓		
Wang, Ma, et al. (2022)		✓		✓		
This paper	Recommended courses to learners so that learners **complete the courses as much as possible.**	✓	✓	✓	✓	✓

7.3.2. Parameter sensitivity analysis for personalized course recommendation using engagement

Tables 3, 5 and 7 show the influence of changing the value of α, Train/Test set ratio, and Batch size on each metric of recommendation accuracy on the XuetangX dataset, where Table 3 shows the influence of changing the value of α on each metric of recommendation accuracy when the Train/Test set ratio is 8:2 and Batch size is 8; Table 5 shows the impact of changing the value of Train/Test set ratio on each metrics of recommendation accuracy when Batch size is 8 and α is 3; Table 7 shows the impact of changing the value of Train/Test set ratio on each metrics of recommendation accuracy when Batch size is 8 and α is 3.

On the KDDCUP dataset, Table 4, Table 6, and Table 8 show the influence of changing the value of α, Train/Test set ratio, and Batch size on each metric of recommendation accuracy, where Table 4 shows the impact of changing the value of α; Table 6 shows the influence of changing the value of Train/Test set ratio; Table 8 shows the impact of changing the value of Batch size.

To sum up, each evaluation metric value of personalized course recommendation fluctuates greatly with the Train/Test set ratio, and Batch size and α have little influence on recommendation results. After extensive experiments, it is known that the model performs best on the XuetangX dataset when the Batch size is 8, $\alpha = 3$, and the training/test set ratio is 8:2. And in the KDDCUP dataset, when the Batch size is 8, $\alpha = 2$, and the Train/Test set ratio is 8:2, using the ENN model to predict the engagement of unenrolled courses for personalized course recommendation performs best.

7.3.3. Baselines for personalized course recommendation using engagement

Table 9 summarizes the baselines used to evaluate recommendation systems and the hit standards in existing work. It can be seen that the majority of course recommendation works' hit standards only focuses on whether learners will be interested in the recommended course, but failed to pay attention to whether learners can complete it after the recommended course. This paper considers this in mind, it adopts engagement to recommend courses, and takes learners' interest in and completion of the recommended courses as our hit standards.

Moreover, through literature research, most of the course recommended works have chosen Popular, Random, MF, CF, and SVD as their baselines, where CF is mostly used for the design of course recommendation (Khalid, Lundqvist, & Yates, 2022), it can be classified into traditional Matrix Factorization (MF)-based methods (Yuan et al., 2021) and deep learning-based methods (Boratto, Fenu, & Marras, 2021; Gómez, Boratto, & Salamó, 2022; He et al., 2017). Deep learning-based methods are the applications of neural networks on recommender systems, and have achieved great success in many application areas. For example, Generalized Matrix Factorization (GMF) applied a linear kernel to model the latent feature interactions, Multi-Layer Perceptron can provide a high level of nonlinear modeling capabilities, He et al. (2017) proposed to add hidden layers on the concatenated vector, using a standard MLP to learn the interaction between user and item latent features. Neural Collaborative Filtering (NCF) model (He et al., 2017) combined GMF with MLP for modeling user-course latent structures, which learns the probability of recommending target courses to related users.

Therefore, the baseline of this paper can be divided into the following three categories, which are non-personalized recommendation methods (Popular, Random), traditional recommendation algorithms (MF, SVD++), and deep learning-based methods (GMF, MLP, NCF). The comparison method is as follows.

- **Non-personalized**
  - **Popular** Recommended courses for which most learners are enrolled.
  - **Random** Randomly chooses one of the candidate courses from the pool.

Table 10
Results of personalized course recommendations using ENN-predicted learner engagement in unenrolled courses and comparing them to baselines using the **XuetangX** dataset. All comparison methods use the hit standards proposed in this paper, where learners are not only interested in the recommended course but can complete it. The best results are highlighted in bold, and the second-best results are underlined.

Methods	Hit@5	Hit@10	Hit@15	R@5	R@10	R@15	P@5	P@10	P@15	F1@5	F1@10	F1@15
Popular	7.55%	14.48%	<u>28.23%</u>	7.66%	14.67%	<u>29.22%</u>	1.68%	1.61%	2.09%	2.71%	2.87%	3.88%
Random	3.04%	5.82%	8.77%	3.04%	5.74%	8.71%	2.26%	4.19%	6.32%	2.48%	4.61%	6.96%
QE+MF	6.11%	8.12%	9.00%	6.74%	8.93%	9.80%	6.68%	8.83%	9.6%	6.70%	8.84%	9.66%
QE+SVD++	2.49%	3.80%	8.56%	2.44%	3.61%	8.40%	2.06%	2.99%	7.22%	2.17%	3.17%	7.57%
QE+GMF	2.11%	3.86%	5.70%	2.05%	3.78%	5.60%	1.81%	3.30%	4.84%	1.88%	3.43%	5.05%
QE+MLP	<u>16.26%</u>	22.18%	25.53%	<u>17.35%</u>	23.59%	26.98%	<u>16.66%</u>	19.89%	22.48%	<u>16.91%</u>	20.21%	22.85%
QE+NCF	16.25%	<u>22.52%</u>	25.96%	17.32%	<u>23.99%</u>	27.50%	16.50%	<u>22.82%</u>	<u>26.02%</u>	16.77%	<u>23.20%</u>	<u>26.49%</u>
QE+ENN	**21.16%**	**25.77%**	**28.88%**	**22.68%**	**27.61%**	**30.90%**	**21.74%**	**26.53%**	**29.73%**	**22.04%**	**26.88%**	**30.11%**

Table 11
Results of personalized course recommendations using ENN-predicted learner engagement in unenrolled courses and comparing them to baselines using the **KDDCUP** dataset. All comparison methods use the hit standards proposed in this paper, where learners are not only interested in the recommended course but can complete it. The best results are highlighted in bold, and the second-best results are underlined.

Methods	Hit@3	Hit@5	Hit@10	R@3	R@5	R@10	P@3	P@5	P@10	F1@3	F1@5	F1@10
Popular	13.32%	24.43%	<u>38.79%</u>	13.33%	24.77%	<u>39.42%</u>	4.97%	5.47%	4.35%	7.11%	8.85%	7.76%
Random	10.89%	17.21%	33.97%	10.99%	17.18%	34.16%	8.60%	13.16%	25.98%	9.34%	14.38%	28.47%
QE+MF	3.61%	5.93%	15.63%	3.24%	5.26%	14.49%	2.61%	4.21%	11.65%	2.79%	4.53%	12.53%
QE+SVD++	8.01%	15.53%	35.85%	7.69%	15.25%	35.56%	6.49%	13.00%	29.82%	6.85%	13.70%	31.61%
QE+GMF	10.35%	17.25%	30.59%	10.08%	16.81%	29.66%	8.52%	14.09%	24.58%	9.01%	14.95%	26.18%
QE+MLP	**19.18%**	<u>24.98%</u>	37.76%	**19.24%**	<u>24.95%</u>	37.55%	<u>12.69%</u>	<u>21.36%</u>	<u>31.73%</u>	**17.56%**	<u>22.51%</u>	<u>33.57%</u>
QE+NCF	12.93%	18.92%	33.22%	12.64%	18.55%	32.46%	10.77%	15.59%	27.12%	11.35%	16.51%	28.79%
QE+ENN	<u>16.15%</u>	**28.04%**	**41.17%**	<u>15.87%</u>	**27.89%**	**41.18%**	**13.47%**	**23.93%**	**35.21%**	<u>14.22%</u>	**25.19%**	**37.11%**

- **Traditional**
  - **MF** Matrix Factorization. Each learner and his enrolled course are embedded in a fixed-length vector. Whether to recommend a course depends on the dot product of the learner vector and course vector.
  - **SVD++** Gradient descent matrix factorization.
- **Deep learning-based**
  - **GMF** Generalized Matrix Factorization, it applies a linear kernel to model the latent feature interactions.
  - **MLP** This paper proposed to add hidden layers on the concatenated vector, using a standard MLP to learn the interaction between user and item latent features.
  - **NCF** It is a hybrid model that combines GMF with MLP for modeling user-course latent structures, which learns the probability of recommending target courses to related users.

QE is integrated into the aforementioned baselines, and the comparison methods can be described as "QE+MF", "QE+SVD++", "QE+GMF", "QE+MLP", and "QE+NCF" respectively. In addition, all comparison methods use the personalized course recommendation described in the Stage 3 of Section 6. Popular only consider the number of learners enrolled in courses as popularity of courses, do not consider the engagement, whereas Random just randomly recommend, it is independent of the engagement.

7.3.4. Experimental results for MOOC recommendation using engagement

Tables 10 and 11 show the comparison of the recommendation results of all models on the two datasets. Since the XuetangX dataset contains 247 courses and the KDDCUP dataset contains 39 courses, this paper chooses different Top-K for the recommendation. For the XuetangX dataset, this paper calculates and compares Top-5, Top-10, and Top-15. For the KDDCUP dataset, calculate and compare Top-3, Top-5, and Top-10 respectively. On the XuetangX dataset, the ENN model has better results compared to other models. We can see that when recommending 5 courses, Hit ratio improves by 4.46% on average, Recall improves by 4.50% on average, Precision improves by 2.30% on average, and F1-score improves by 4.75% on average. When recommending 10 courses, it showed a mean improvement of 2.65% in Hit ratio, 3.95% in Recall, 4.67% in Precision, 4.53% in F1-score. The average improvement is 3.14% in Hit@15, 3.24% in R@15, 3.24% in P@15, and 4.26% in F1@15.

On the KDDCUP dataset, when recommending 5 courses, The MLP model is slightly higher than the ENN in all three metrics, but the rest of the metrics are all lower than ENN. Overall, the ENN model has better results in terms of Recall, Hit ratio, Precision, and F1-score compared to baselines (Popular, Random, MF, SVD++, GMF, MLP, NCF). This phenomenon occurs because the ENN model takes into account the rich course information of the learner, which makes the ENN have good predictive accuracy, so that the engagement prediction results closely resemble the learners' actual engagement within the course. Therefore, using the ENN model to predict learner engagement on unenrolled courses and personalizing course recommendations based on it outperforms baselines.

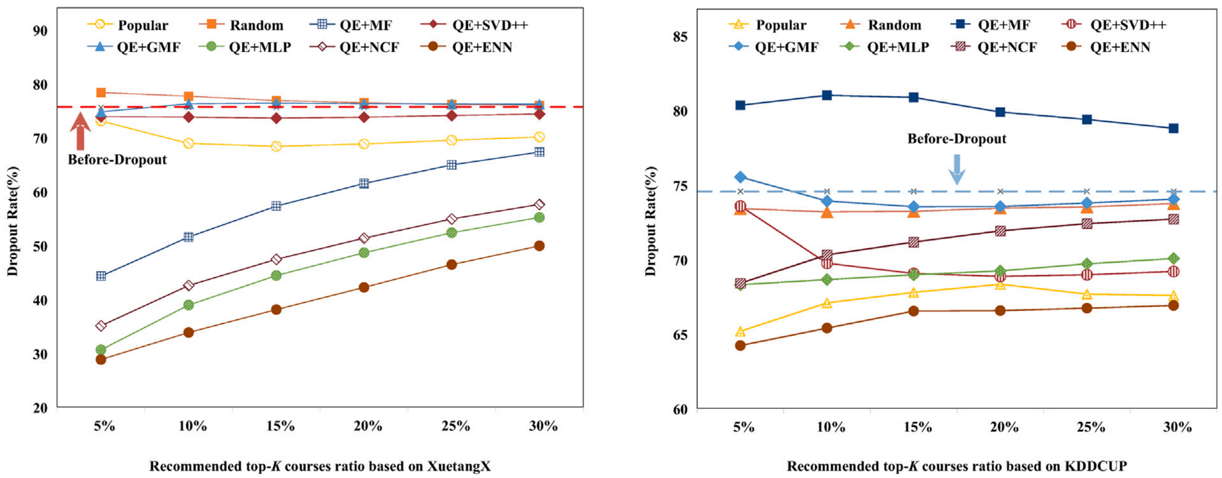


Fig. 13. The relationship between dropout rates and the number of recommended courses after applying various recommendation models. The x-axis represents recommended top-K courses ratio, K is taken as 5%, 10%, 15%, 20%, 25%, and 30% respectively. The y-axis indicates the dropout rate of the MOOC platform after adopting the recommendation strategy. In the left panel, the red dashed line represents the dropout rate observed in the XuetangX dataset before any course recommendations are made, while the blue dashed line in the right panel represents the initial dropout rate in the KDDCUP dataset before recommendations. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

Table 12
The impact of the components on ENN model predictions.

Method	XuetangX		KDDCUP	
	RMSE	MAE	RMSE	MAE
ENN-f (no $\mathbf{f}'$)	0.1085	0.0744	0.0635	0.0409
ENN-g (no $\mathbf{g}'$)	0.1321	**0.0705**	0.0709	**0.0348**
ENN	**0.1065**	0.0726	**0.0624**	0.0396

7.4. Theoretical and practical implications: Reducing dropout rate for MOOC platforms using predicted engagement

In addition, this paper does an interesting experiment. Using engagement to personalized course recommendations to learners can reduce the dropout rates of MOOC platforms. The formula is as follows,

$$Before-Dropout = \frac{\sum_{u_i \in U} |D(u_i)|}{\sum_{u_i \in U} \left|C\left(u_i\right)^+\right|}, \qquad After-Dropout = 1 - \frac{\sum_{u_i \in U} |\mathrm{PCR}@K\left(u_i\right) \cap \left(C\left(u_i\right)^+ - D\left(u_i\right)\right)|}{\sum_{u_i \in U} |C\left(u_i\right)^+ \cap \mathrm{PCR}@K\left(u_i\right)|}, \tag{24}$$

where $\mathrm{PCR}@K\left(u_i\right) \cap \left(C\left(u_i\right)^+ - D\left(u_i\right)\right)$ is the set of recommended top-K courses to u_i and which he completed. $C\left(u_i\right)^+ \cap \mathrm{PCR}@K\left(u_i\right)$ is the set of the top-K courses recommended to u_i and enrolled by him.

From Eq. (24), it can be concluded that the course dropout rate of the MOOC platform reaches 75.78% at the beginning of the XuetangX dataset and 74.55% on the KDDCUP. Fig. 13 illustrates the relationship between dropout rates and the number of recommended courses after applying various recommendation models. Notably, when recommending a 5% high engagement course in the XuetangX dataset, all compared methods effectively reduce dropout rates, except for the Random method, which may lead to higher dropout rates. The lowest dropout rates were observed when recommending unenrolled courses using the engagement predicted by the ENN model. The ENN model's dropout rate was on average 25.58% lower than the other recommendation models when recommending a 15% high engagement course. On the KDDCUP dataset, when recommending 5% high engagement courses, the MF and GMF approaches to reducing dropout rates are not ideal, but the ENN model proposed in this paper still reduces the dropout rate the best, from the initial 74.55% to 64.21%, when making recommendations.

In addition, we found an interesting phenomenon that the dropout rate tends to increase with the increase in the number of recommended courses, which suggests that the course recommendation should not be too much, otherwise it may lead to an increase in the learning burden of the learners. The experimental results indicate that the utilization of the ENN model, as introduced in this paper, for predicting learner engagement in unenrolled courses and delivering personalized course recommendations significantly reduces dropout rates on MOOC platforms. This result carries substantial implications for enhancing the overall effectiveness and long-term viability of MOOC platforms.

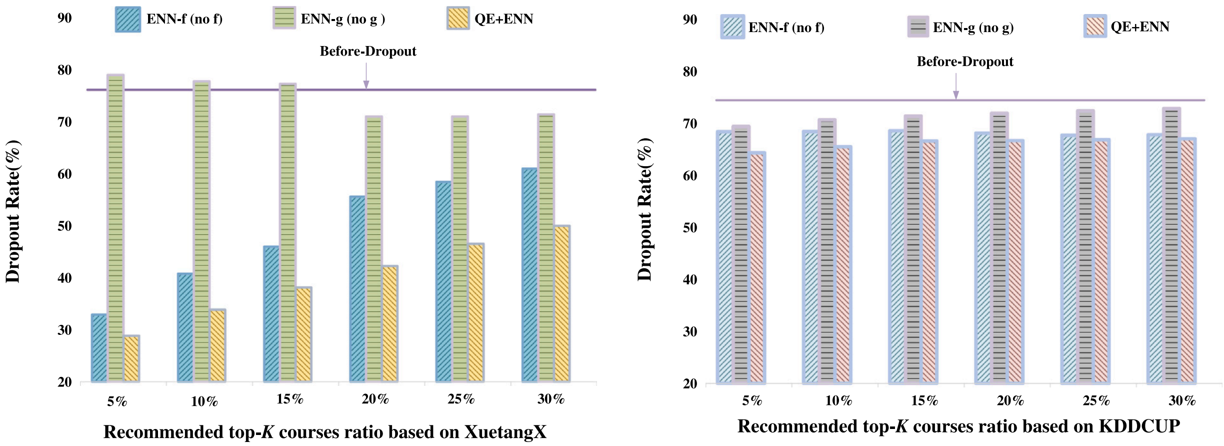


Fig. 14. It shows the effect of the different models in reducing the dropout rate on the two datasets, where the purple horizontal line signifies the MOOC dropout rate prior to the implementation of course recommendations. The x-axis represents the recommended top-K courses ratio, K is taken as 5%, 10%, 15%, 20%, 25%, and 30% respectively. The y-axis indicates the dropout rate of the MOOC platform after adopting the recommendation strategy. (For interpretation of the references to color in this figure legend, the reader is referred to the web version of this article.)

7.5. Component analysis and feature analysis for ENN

In predicting learner engagement in unenrolled courses using ENN, we added two nonlinear layers $\mathbf{f}'$ and $\mathbf{g}'$ which were used to correct the model's predictions, and in order to assess the impact of these two components, we conducted an ablation study by removing one of them to show their role in engagement prediction. The definition after removing a component separately is as follows:

ENN-f (no $\mathbf{f}'$): This is to show the importance of considering $\mathbf{f}'$ components, that is, to predict learner engagement in unenrolled courses by removing $\mathbf{f}'$ from the ENN.

ENN-g (no $\mathbf{g}'$): This is to evaluate the importance of $\mathbf{g}'$ in predicting learner engagement in unenrolled courses, that is, we keep only one component, $\mathbf{f}'$, and remove component $\mathbf{g}'$ from the ENN model.

The experimental results are shown in Table 12, at the same time, we tested the effect of removing a component on the reduction of dropout rates, and the results are shown in Fig. 14. Table 12 shows that the removal of any one of the components $\mathbf{f}'$ and $\mathbf{g}'$ has an impact on the predictive ability of the model, especially the $\mathbf{g}'$ component has a greater impact on the model, and we can see that although the effect of MAE is better than that of the ENN model, the difference in the error of the RMSE is a little bit larger, which reflects that there is a large error between the true value and the predicted value when using the model prediction using ENN-g. In addition, Fig. 14 shows the effect of the different models in reducing the dropout rate on the two datasets, where the purple horizontal line signifies the MOOC dropout rate prior to the implementation of course recommendations. It can be seen that ENN reduces the dropout rate better than ENN-f and ENN-g.

8. Conclusion

Firstly, this paper proposes a method to Quantify Engagement (QE), which effectively quantifies the engagement of learners' enrolled courses according to their learning action data and learning duration. Through experiments, we find that the defined engagement is closely related to the dropout rate. Secondly, a prediction model of Engagement Neural Network (ENN) was proposed, the model can predict before learners enroll for a course of online learning participate, it would assist MOOC platforms in gauging their willingness to continue learning in other unenrolled courses. Finally the relationship between engagement and course dropout rate is used to make personalized course recommendations to learners to ensure that the recommended courses are likely to be completed by learners, thus effectively reducing the dropout rate of MOOC platforms.

In summary, this paper represents an initial exploration into the use of personalized course recommendations as a means to mitigate dropout rates, without exploring alternative strategies for dropout reduction. In our future work, we will try to consider other characteristics such as friends' information, course attribute information, and learners' backgrounds in order to proactively predict dropout and implement early interventions to further diminish dropout rates on MOOC platforms.

CRediT authorship contribution statement

Shu Li: Data curation, Investigation, Software, Visualization, Conceptualization, Methodology, Experimental report, Writing – review & editing. **Yuan Zhao:** Experimental report, Writing – review & editing. **Longjiang Guo:** Funding acquisition, Supervision, Formal analysis, Conceptualization, Experimental design, Project administration, Writing – review & editing. **Meirui Ren:** Supervision, Formal analysis, Validation, Project administration, Writing – review & editing. **Jin Li:** Validation, Heuristic Discussion, Writing – review & editing. **Lichen Zhang:** Funding acquisition, Writing – review & editing. **Keqin Li:** Conceptualization, Validation, Writing – review & editing.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

We have shared the link of http://moocdata.cn/data/user-activity to the used data set in the manuscript.

Acknowledgments

This work is partly supported by the National Natural Science Foundation of China under Grant No. 61977044, 62077035, and 62206162; the Second Batch of New Engineering Research and Practice Projects of the Ministry of Education of China under Grant No. E-RGZN20201045; the Natural Science Basis Research Plan in Shaanxi Province of China under Grant No. 2020JM-302, 2020JM-303 and 2022JM-371; the Key R&D Program of Shaanxi Province Grant No. 2020ZDLGY-10-05; the Fundamental Research Funds for the Central Universities, China Grant No. GK202205037; the Ministry of Education's Cooperative Education Project 202102591018; the CCF-Tencent Open Fund Grant No. RAGR20220127; the Shaanxi Normal University Graduate Pilot Talent Fund Program Grant No. LHRCCX23204.

References

Almutairi, F., & White, S. (2018). How to measure student engagement in the context of blended-MOOC. *Interactive Technology and Smart Education*, *15*(3), 262–278. http://dx.doi.org/10.1108/ITSE-07-2018-0046.

Ban, Q., Wu, W., Hu, W., Lin, H., Zheng, W., & He, L. (2022). Knowledge-enhanced multi-task learning for course recommendation. In *Proceedings of database systems for advanced applications - 27th international conference* (pp. 85–101). Springer, http://dx.doi.org/10.1007/978-3-031-00126-0_6.

Boratto, L., Fenu, G., & Marras, M. (2019). The effect of algorithmic bias on recommender systems for massive open online courses. In *Proceedings of lecture notes in computer science* (pp. 457–472). Springer, http://dx.doi.org/10.1007/978-3-030-15712-8_30.

Boratto, L., Fenu, G., & Marras, M. (2021). Connecting user and item perspectives in popularity debiasing for collaborative recommendation. *Information Processing & Management*, *58*(1), Article 102387. http://dx.doi.org/10.1016/j.ipm.2020.102387.

Borrella, I., Caballero-Caballero, S., & Ponce-Cueto, E. (2022). Taking action to reduce dropout in MOOCs: Tested interventions. *Computer Education*, *179*, Article 104412. http://dx.doi.org/10.1016/j.compedu.2021.104412.

Cheng, Y., Nunes, B. P., & Manrique, R. (2022). Not another hardcoded solution to the student dropout prediction problem: A novel approach using genetic algorithms for feature selection. In *Intelligent tutoring systems - 18th international conference, ITS 2022, Bucharest, Romania, June 29 - July 1, 2022, proceedings, vol. 13284* (pp. 238–251). Springer, http://dx.doi.org/10.1007/978-3-031-09680-8_23.

Cole, A. W., Lennon, L., & Weber, N. L. (2021). Student perceptions of online active learning practices and online learning climate predict online course engagement. *Interactive Learning Environments*, *29*(5), 866–880. http://dx.doi.org/10.1080/10494820.2019.1619593.

D'Aniello, G., de Falco, M., Gaeta, M., & Lepore, M. (2020). Feedback generation using fuzzy cognitive maps to reduce dropout in situation-aware e-learning systems. In *IEEE conference on cognitive and computational aspects of situation management* (pp. 195–199). IEEE, http://dx.doi.org/10.1109/CogSIMA49017.2020.9216177.

Drousiotis, E., Pentaliotis, P., Shi, L., & Cristea, A. I. (2021). Capturing fairness and uncertainty in student dropout prediction - A comparison study. In *Artificial intelligence in education - 22nd international conference, AIED 2021, Utrecht, the Netherlands, June 14-18, 2021, proceedings, part II, vol. 12749* (pp. 139–144). Springer, http://dx.doi.org/10.1007/978-3-030-78270-2_25.

Feng, W., Tang, J., & Liu, T. X. (2019). Understanding dropouts in MOOCs. In *Proceedings of the thirty-third AAAI conference on artificial intelligence* (pp. 517–524). AAAI, http://dx.doi.org/10.1609/aaai.v33i01.3301517.

Gómez, E., Boratto, L., & Salamó, M. (2022). Provider fairness across continents in collaborative recommender systems. *Information Processing & Management*, *59*(1), Article 102719. http://dx.doi.org/10.1016/j.ipm.2021.102719.

Hancox, Z., & Relton, S. D. (2022). Temporal graph-based CNNs (TG-CNNs) for online course dropout prediction. In *Foundations of intelligent systems - 26th international symposium, ISMIS 2022, Cosenza, Italy, October 3-5, 2022, proceedings, vol. 13515* (pp. 357–367). Springer, http://dx.doi.org/10.1007/978-3-031-16564-1_34.

He, X., Liao, L., Zhang, H., Nie, L., Hu, X., & Chua, T. (2017). Neural collaborative filtering. In *Proceedings of the 26th international conference on world wide web* (pp. 173–182). ACM, http://dx.doi.org/10.1145/3038912.3052569.

Huang, A. Y. Q., Lu, O. H. T., & Yang, S. J. H. (2023). Effects of artificial intelligence-enabled personalized recommendations on learners' learning engagement, motivation, and outcomes in a flipped classroom. *Computer Education*, *194*, Article 104684. http://dx.doi.org/10.1016/j.compedu.2022.104684.

Jing, X., & Tang, J. (2017). Guess you like: course recommendation in MOOCs. In *Proceedings of the international conference on web intelligence* (pp. 783–789). ACM, http://dx.doi.org/10.1145/3106426.3106478.

Jung, H., Jang, Y., Kim, S., & Kim, H. (2022). KPCR: knowledge graph enhanced personalized course recommendation. In *Proceedings of AI 2021: Advances in artificial intelligence - 34th Australasian joint conference* (pp. 739–750). Springer, http://dx.doi.org/10.1007/978-3-030-97546-3_60.

Khalid, A., Lundqvist, K., & Yates, A. (2022). A literature review of implemented recommendation techniques used in Massive Open online Courses. *Expert Systems with Applications*, *187*(September 2021), Article 115926. http://dx.doi.org/10.1016/j.eswa.2021.115926.

Le, T., Vo, V., Nguyen, K., & Le, B. (2020). Improving deep matrix factorization with normalized cross entropy loss function for graph-based MOOC recommendation. In *Proceedings of proc. int. conf. comput. graph., vis., comput. vis. image process* (pp. 141–148). IEEE, http://dx.doi.org/10.33965/bigdaci2020_202011l017.

Li, D., Liu, H., Zhang, Z., Lin, K., Fang, S., Li, Z., et al. (2021). CARM: confidence-aware recommender model via review representation learning and historical rating behavior in the online platforms. *Neurocomputing*, *455*, 283–296. http://dx.doi.org/10.1016/j.neucom.2021.03.122.

Lin, Y., Feng, S., Lin, F., Zeng, W., Liu, Y., & Wu, P. (2021). Adaptive course recommendation in MOOCs. *Knowledge-Based Systems*, *224*, Article 107085. http://dx.doi.org/10.1016/j.knosys.2021.107085.

Lin, Y., Lin, F., Zeng, W., Xiahou, J., Li, L., Wu, P., et al. (2022). Hierarchical reinforcement learning with dynamic recurrent mechanism for course recommendation. *Knowledge-Based Systems*, *244*(May), Article 108546. http://dx.doi.org/10.1016/j.knosys.2022.108546.

Lin, J., Sun, G., Shen, J., Pritchard, D., Yu, P., Cui, T., et al. (2021). MOOC student dropout rate prediction via separating and conquering micro and macro information. In *Neural information processing - 28th international conference, ICONIP 2021, Sanur, Bali, Indonesia, December 8-12, 2021, proceedings, part VI, vol. 1517* (pp. 459–467). Springer, http://dx.doi.org/10.1007/978-3-030-92310-5_53.

Ma, J., Han, X., Yang, J., & Cheng, J. (2015). Examining the necessary condition for engagement in an online learning environment based on learning analytics approach: The role of the instructor. *Internet and Higher Education*, *24*, 26–34. http://dx.doi.org/10.1016/j.iheduc.2014.09.005.

Ma, D., Wang, Y., Chen, M., & Shen, J. (2021). SRACR: Semantic and relationship-aware online course recommendation. In *Proceedings of 2021 IEEE international conference on engineering, technology &education* (pp. 367–374). IEEE, http://dx.doi.org/10.1109/TALE52509.2021.9678921.

Mehta, N. K., Prasad, S. S., Saurav, S., Saini, R., & Singh, S. (2022). Three-dimensional DenseNet self-attention neural network for automatic detection of student's engagement. *Applied Intelligence*, *52*(12), 13803–13823. http://dx.doi.org/10.1007/s10489-022-03200-4.

Mogavi, R. H., Ma, X., & Hui, P. (2021). Characterizing student engagement moods for dropout prediction in question pool websites. *Proceedings of the ACM Human-Computer Interaction*, *5*(CSCW1), 12:1–12:22. http://dx.doi.org/10.1145/3449086.

Molina, O. M. A., Obando, G. L., Bastidas, A. A., & Mosquera, E. M. (2022). Motivation: Personalization of contents to reduce dropout rates. In *IEEE learning with MOOCS* (pp. 61–66). IEEE, http://dx.doi.org/10.1109/LWMOOCS53067.2022.9927849.

Obeidat, R., Duwairi, R., & Al-Aiad, A. (2019). A collaborative recommendation system for online courses recommendations. In *Proceedings of 2019 international conference on deep learning and machine learning in emerging applications* (pp. 49–54). IEEE, http://dx.doi.org/10.1109/Deep-ML.2019.00018.

O'Brien, H. L., Arguello, J., & Capra, R. (2020). An empirical study of interest, task complexity, and search behaviour on user engagement. *Information Processing & Management*, *57*(3), Article 102226. http://dx.doi.org/10.1016/j.ipm.2020.102226.

Plak, S., van Klaveren, C., & Cornelisz, I. (2023). Raising student engagement using digital nudges tailored to students' motivation and perceived ability levels. *British Journal of Educational Technology*, *54*(2), 554–580. http://dx.doi.org/10.1111/bjet.13261.

Raguro, M. C. F., Lagman, A. C., Abad, L. P., & Ong, P. L. S. (2022). Extraction of LMS student engagement and behavioral patterns in online education using decision tree and K-means algorithm. In *APIT 2022: 4th Asia Pacific information technology conference* (pp. 138–143). ACM, http://dx.doi.org/10.1145/3512353.3512373.

Sakboonyarat, Siriporn, Tantatsanawong, & Panjai (2019). Massive open online courses (MOOCs) recommendation modeling using deep learning. In *Proceedings of 2019 23rd international computer science and engineering conference* (pp. 275–280). IEEE, http://dx.doi.org/10.1109/ICSEC47112.2019.8974770.

Shao, E., Guo, S., & Pardos, Z. A. (2021). Degree planning with PLAN-BERT: multi-semester recommendation using future courses of interest. In *Proceedings of thirty-fifth AAAI conference on artificial intelligence* (pp. 14920–14929). AAAI, URL: https://ojs.aaai.org/index.php/AAAI/article/view/17751.

Soffer, T., & Cohen, A. (2019). Students' engagement characteristics predict success and completion of online courses. *Journal of Computer Assisted Learning*, *35*(3), 378–389. http://dx.doi.org/10.1111/jcal.12340.

Song, D., Rice, M., & Oh, E. Y. (2019). Participation in online courses and interaction with a virtual agent. *International Review of Research in Open and Distributed Learning*, *20*(1), 44–62. http://dx.doi.org/10.19173/irrodl.v20i1.3998.

Symeonidis, P., Malakoudis, D., et al. (2019). Multi-modal matrix factorization with side information for recommending massive open online courses. *Expert Systems with Applications*, *118*(MAR.), 261–271. http://dx.doi.org/10.1016/j.eswa.2018.09.053.

Tian, X., & Liu, F. (2021). Capacity tracing-enhanced course recommendation in MOOCs. *IEEE Transactions on Learning Technologies*, *14*(3), 313–321. http://dx.doi.org/10.1109/TLT.2021.3083180.

Wang, X., Ma, W., Guo, L., Jiang, H., Liu, F., & Xu, C. (2022). HGNN: Hyperedge-based graph neural network for MOOC course recommendation. *Information Processing & Management*, *59*(3), Article 102938. http://dx.doi.org/10.1016/j.ipm.2022.102938.

Wang, C., Zhu, H., Wang, P., Zhu, C., Zhang, X., Chen, E., et al. (2022). Personalized and explainable employee training course recommendations: A Bayesian variational approach. *ACM Transactions on Information Systems*, *40*(4), 70:1–70:32. http://dx.doi.org/10.1145/3490476.

Xing, W. D. (2019). Dropout prediction in MOOCs: Using deep learning for personalized intervention. *Journal of Educational Computing Research*, *57*, http://dx.doi.org/10.1177/0735633118757015.

Xu, G., Jia, G., Shi, L., & Zhang, Z. (2021). Personalized course recommendation system fusing with knowledge graph and collaborative filtering. *Computational Intelligence and Neuroscience*, *2021*, 9590502:1–9590502:8. http://dx.doi.org/10.1155/2021/9590502.

Yang, S., & Cai, X. (2022). Bilateral knowledge graph enhanced online course recommendation. *Information Systems*, *107*, Article 102000. http://dx.doi.org/10.1016/j.is.2022.102000.

Yao, W., Sun, H., & Hu, X. (2020). A novel search ranking method for MOOCs using unstructured course information. *Wireless Communications and Mobile Computing*, *2020*, 8813615:1–8813615:13. http://dx.doi.org/10.1155/2020/8813615.

Yuan, X., Han, L., Qian, S., Zhu, L., Zhu, J., & Yan, H. (2021). Preliminary data-based matrix factorization approach for recommendation. *Information Processing & Management*, *58*(1), Article 102384. http://dx.doi.org/10.1016/j.ipm.2020.102384.

Zhang, H., Huang, T., Lv, Z., Liu, S., & Zhou, Z. (2018). MCRS: A course recommendation system for MOOCs. *Multimedia Tools and Applications*, *77*(6), 7051–7069. http://dx.doi.org/10.1007/s11042-017-4620-2.

Zhang, J., & Ma, K. (2022). EDPS: Early dropout prediction system of MOOC courses. In *29th Asia-Pacific software engineering conference* (pp. 562–563). IEEE, http://dx.doi.org/10.1109/APSEC57359.2022.00081.

Zhang, H., Shen, X., Yi, B., Wang, W., & Feng, Y. (2023). KGAN: Knowledge grouping aggregation network for course recommendation in MOOCs. *Expert Systems with Applications*, *211*, Article 118344. http://dx.doi.org/10.1016/j.eswa.2022.118344.