Article

A Knowledge Concept Recommendation Model Based on Tensor Decomposition and Transformer Reordering

Zhaoyu Shou [1], Yishuai Chen [1], Hui Wen [1,*], Jinghua Liu [1], Jianwen Mo [1] and Huibing Zhang [2]

1   School of Information and Communication, Guilin University of Electronic Technology, Guilin 541004, China; guilinshou@guet.edu.cn (Z.S.); 20022303014@mails.guet.edu.cn (Y.C.); ddy2003@guet.edu.cn (J.L.); jwmo@guet.edu.cn (J.M.)
2   School of Computer and Information Security, Guilin University of Electronic Technology, Guilin 541004, China; zhanghuibing@guet.edu.cn
*   Correspondence: huiwen@guet.edu.cn

Abstract: To help students choose the knowledge concepts that meet their needs so that they can learn courses in a more personalized way, thus improving the effectiveness of online learning, this paper proposes a knowledge concept recommendation model based on tensor decomposition and transformer reordering. Firstly, the student tensor, knowledge concept tensor, and interaction tensor are created based on the heterogeneous data of the online learning platform are fused and simplified as an integrated tensor; secondly, we perform multi-dimensional comprehensive analysis on the integrated tensor with tensor-based high-order singular value decomposition to obtain the student personalized feature matrix and the initial recommendation sequence of knowledge concepts, and then obtain the latent embedding matrix of knowledge concepts via Transformer that combine initial recommendation sequence of knowledge concepts and knowledge concept learning sequential information; finally, the final Top-N knowledge concept recommendation list is generated by fusing the latent embedding matrix of knowledge concepts and the students' personalized feature matrix. Experiments on two real datasets show that the model recommendation performance of this paper is better compared to the baseline model.

Keywords: knowledge concept recommendation; tensor decomposition; transformer reordering; online learning

1. Introduction

In recent years, the growth of online learning platforms like MOOCs and the gradual shift from face-to-face learning to online teaching has brought online learning to the forefront of attention [1]. Online learning platforms are cumbersome in terms of learning resources and prone to problems such as disorientation and information overload [2]. To improve the learning experience for students and to enhance learning and teaching effectiveness, a large number of researchers are working on personalized recommendations for students [3,4]. Jena et al. proposed a collaborative filtering-based recommendation system for recommending e-learning courses to learners as a way to help them choose e-learning courses based on their preferences [5]. Shen et al. proposed an online course recommendation model based on an autoencoder, which was improved using long-term and short-term memory (LSTM) networks to extract temporal features of the data [6]. Zhu et al. proposed a hybrid recommendation model incorporating network structure features with graph neural networks and user interaction activities with tensor decomposition to use heterogeneous features for course recommendation [7]. Pu et al. proposed an exercise recommendation algorithm based on cognitive level and data mining to further meet the personalized needs of students [8]. Diao et al. proposed a personalized learning path recommendation method based on weak concept mining, which improved students' learning experience and learning outcomes [9]. Liu et al. proposed a transformer-based learning

path recommendation system to bridge the gap in learning resource recommendations [10]. The existing recommendation models mainly focus on course resource recommendations; however, courses in an online learning platform often consist of multiple videos, and a video may contain more than one knowledge concept; therefore, simply recommending courses will ignore students' interests in specific knowledge concepts [11], and the interest of different students in the same course is also variable. Therefore, this paper considers improving students' online learning experience from the perspective of knowledge concept recommendation. In a related study on knowledge concepts, Gong et al. proposed an attention-based graph convolutional network that can effectively mine and aggregate users' potential interests as a way to make knowledge concept recommendations [11]. Zhao et al. extracted concept-level prerequisite relations and course prerequisite relations from MOOC titles and embedded the inter-course prerequisite relations in a neural attention network to implement course recommendations [12]. However, these models have certain limitations: firstly, the student, knowledge concept, and interaction information data contained in an online course exhibit a multidimensional multivariate interrelated structure, and these approaches extract only a portion of the student information from the online learning process for modeling purposes, failing to retain the information integrity in the high-dimensional space and losing some of the latent semantic association information between the data; secondly, they fail to take into account the importance of sequential information in the recommendation of knowledge concepts.

Based on the above analysis, this paper proposes a tensor decomposition and transformer reordering-based knowledge concept recommendation model (TTRKRec). Firstly, the student tensor, knowledge concept tensor and interaction tensor created based on the heterogeneous data from the online learning platform are fused and simplified into a composite tensor to maintain the heterogeneous relevance of the data; secondly, the tensor-based higher-order singular value method is used to obtain the student personalized feature matrix and the initial recommended sequence of knowledge concepts by multi-dimensional synthesis analysis of the integrated tensor; then, the transformer is used to combine the initial recommended sequence of knowledge concepts and the knowledge concept learning sequential information to achieve the latent embedding matrix of knowledge concepts; finally, the latent embedding matrix and the student's personalized feature matrix are fused to complete the knowledge concept sequence recommendation and generate the final Top-N knowledge concept recommendation list.

The contributions of this paper are as follows:

1. The integrity of the heterogeneous data of the online course is preserved by modeling the student, knowledge concept, and interaction data through the creation of a tensor, and the overall data are analyzed comprehensively in multiple (student, knowledge concept, cognitive level, knowledge concept achievement, and student–system interaction) dimensions using a tensor-based higher-order singular value decomposition to uncover latent information between the data.
2. The transformer encoder layer is used to capture sequential information between knowledge concepts and to fusion personalized student characteristics, enabling more accurate knowledge concept recommendations.
3. Extensive experiments are conducted on two real datasets, and the experimental results demonstrate the advantages of the TTRKRec proposed in this paper compared to several state-of-the-art knowledge concept recommendation models.

The rest of the thesis is organized as follows: Section 2 presents research related to knowledge concept recommendation. Section 3 describes the definitions and computations associated with the model. Section 4 proposes a model for knowledge concept recommendation based on tensor decomposition and transformer reordering. Section 5 presents a comparative analysis of different experimental results to evaluate the performance of the model. Finally, Section 6 concludes the work.

2. Related Work

Currently, knowledge graph-based recommendation methods are commonly used in research related to knowledge concepts. For example, Shi et al. proposed a multidimensional knowledge graph-based learning path recommendation model, which enables the recommended learning path to better meet the learning needs of learners [13]. Wang et al. proposed a framework for a learning path discovery system based on knowledge graphs and DE algorithms, which utilizes subject knowledge graphs in finance to meet the needs of personalized learning path discovery and resource recommendation [14]. Huang et al. proposed a collaborative user filtering recommendation algorithm based on knowledge graphs, which mitigates the user cold-start problem in collaborative filtering algorithms by exploiting the rich semantic relationships in knowledge graphs [15]. Yang et al. combined knowledge graphs with recommender systems to provide a large amount of additional auxiliary information for personalized recommender systems, effectively alleviating the cold start problem [16]. Knowledge graph-based recommendation systems can make the recommendation results interpretable but suffer from the problem of missing relationships or entities, which leads to the deterioration of the recommendation results.

To better mine heterogeneous data relationships and auxiliary information and improve recommendation accuracy, existing research has focused on how to use heterogeneous graph and graph neural network-based approaches for student and educational resource information mining. Wang et al. proposed a knowledge concept recommendation model based on heterogeneous information networks and used the Skip-gram model of Gumbel-Softmax to learn the representation of entities from the generated multifaceted sequences, enriching the contextual semantics of nodes and saving space consumption [17]. Zhao et al. proposed an enhanced knowledge concept recommendation model with heterogeneous information networks, which optimizes the representation of sparse data users by automatically identifying valid meta-paths and multi-hop connections [18]. Ye et al. propose a knowledge concept recommendation model based on heterogeneous information networks and graph convolution that can consider information about the community structure as well as information about node neighborhoods [19]. Heterogeneous information networks enable a more accurate representation of students and knowledge concepts, but rely too much on the similarity of meta-paths, failing to fully explore the underlying characteristics of students and knowledge concepts. Ling et al. proposed a knowledge concept recommendation model based on structurally augmented interactive graph neural networks to help students learn better online, constructing all user knowledge concept history interaction sequences into a knowledge concept interaction graph based on learning paths, and further improving the representation of knowledge concepts using an attention mechanism [20]. Liang et al. proposed a learning resource recommendation method based on graph convolutional networks and reinforcement learning, which embeds multiple paths and performs a student-centered search to improve recommendation accuracy [21]. However, graph neural networks are powerful but require some complex design to be applied to heterogeneous information. Some studies have also introduced tensor decomposition into the field of personalized recommendation. Sun et al. proposed a tensor decomposition model based on label regularization, which incorporates social information to improve the quality of recommendations [22]. Hong et al. proposed a fifth-order tensor model consisting of user, item, multiple ratings, and spatiotemporal data to reflect multi-level spatial and temporal information into the recommendation service, which effectively improves the recommendation performance of multi-criteria recommendation systems [23]. Liu et al. proposed a correlation analysis and personalized recommendation algorithm based on incremental tensor from multiple dimensions of global education data, which can recommend suitable resources in different contexts and has high recommendation performance [24]. Thus, the use of tensor decomposition helps to discover hidden structures and values from the massive amount of data. Based on the above studies, a summary of the relevant research models is shown in Table 1.

Table 1. Summary of relevant research models.

Models	Paper Numbers	Advantages	Limitations
Knowledge graph-based recommendation models	[13–16]	Making recommendations interpretable	Problems with missing relationships or entities
Recommendation models based on heterogeneous information networks	[17–19]	Achieves a more accurate representation of students and knowledge concepts	Over-reliance on meta-path similarity
Graphical neural-based recommendation models	[11,20,21]	High ability to extract time-series features	Requires some complex design to apply to heterogeneous information
Recommendation models based on tensor decomposition	[22–24]	Suitable for the representation and extraction of potential features in high-dimensional data spaces	Weak ability to capture semantic and sequential information

In summary, most existing conceptual knowledge recommendation systems only recommend content and resources for learners at the beginning of the learning process that is likely to match their learning interests, and ignore other supporting information during the learning process that affects learning outcomes, such as interaction behavior, learning performance, and cognitive level. More importantly, the existing knowledge concept recommendation models do not consider the influence of knowledge concept sequential information on the recommendation results. In addition, tensor as a representation model for heterogeneous data is conducive to the mining of implicit information in the data. Therefore, this paper considers more auxiliary information based on tensor modeling to ensure the integrity of heterogeneous data and uses a transformer to fuse knowledge concept sequential information and students' personalized characteristics to achieve knowledge concept recommendation sequence rearrangement.

3. Correlation Definition

In this section, the proposed model's relevant definitions and computational methods are described, and some of the definitions are analyzed and illustrated.

3.1. The Degree of Student–System Interaction

The degree of student–system interaction can be defined as the workload of students watching knowledge concept videos [25]. The degree of student–system interaction $SS_{u,k}$ can be expressed in Equation (1).

$$SS_{u,k} = \alpha_1 \times f_{SS_{u,k}} + \alpha_2 \times t_{SS_{u,k}} + \alpha_3 \times p_{SS_{u,k}} \tag{1}$$

In Equation (1), $f_{SS_{u,k}}$ represents the frequency with which the student u learns knowledge concept k, $t_{SS_{u,k}}$ represents the duration of the student u who learns knowledge concept k and $p_{SS_{u,k}}$ represents the frequency of pausing and dragging with which the student u learns knowledge concept k. According to the literature [26], it was concluded that the degree of student–system interaction was best characterized when $(\alpha_1, \alpha_2, \alpha_3) = (1, 5, 4)$.

3.2. The Degree of Student–Teacher Interaction

The degree of student–teacher interaction is measured by the amount of student–teacher interaction [25], which depends primarily on the question-and-answer process between the teacher and the student. The degree of student–teacher interaction $ST_{u,k}$ can be expressed in Equation (2).

$$\begin{cases} ST_{u,k} = W_{ST_{u,k}} \times f_{ST_{u,k}} \\ W_{ST_{u,k}} = \dfrac{t_{ST_{u,k}}}{\max\left\{ t_{ST_{1,k}}, t_{ST_{2,k}}, \cdots, t_{ST_{a,k}} \right\}} \end{cases} \tag{2}$$

In Equation (2), $W_{ST_{u,k}}$ represents the weighting factor of student–teacher interaction for the student u for knowledge concept k, $f_{ST_{u,k}}$ represents the frequency of student–teacher interaction for the student u for knowledge concept k, a is the total number of students taking the online course at each grade level, $t_{ST_{u,k}}$ represents the duration of student–teacher interaction for the student u for knowledge concept k, and $\max\left\{t_{ST_{1,k}}, t_{ST_{2,k}}, \cdots, t_{ST_{a,k}}\right\}$ is the maximum duration of student–teacher interaction for knowledge concept k.

3.3. Tensor and Tensor Calculations

In an Nth-order tensor $T \in R^{I_1 \times I_2 \times \cdots \times I_N}$, N is the order of the tensor and $I_n (1 \leq n \leq N)$ is the dimension of the nth order of the tensor T.

Definition 1. *Tensor join [27]. Given two tensors $T_1 \in R^{I_1 \times I_2 \times \cdots \times I_M \times K_1 \times K_2 \times \cdots \times K_Q}$ and $T_2 \in R^{K_1 \times K_2 \times \cdots \times K_Q \times J_1 \times J_2 \times \cdots \times J_N}$, with a common mode $K_1, K_2, \cdots, K_Q$, T_1 and T_2 are connected by a tensor to produce a new tensor $T_{new} \in R^{I_1 \times I_2 \times \cdots \times I_M \times J_1 \times J_2 \times \cdots \times J_N \times K_1 \times K_2 \times \cdots \times K_Q}$ whose elements are the product of the elements of T_1 and the elements of T_2.*

Definition 2. *Tensor Simplification [24]. Given an Nth-order tensor $T \in R^{I_1 \times I_2 \times \cdots \times I_N}$, the tensor simplification from tensor T along the pth, $\cdots$, qth orders yields a new tensor $T' \in R^{I_p \times \cdots \times I_q}$ with elements $t'_{i_p, \cdots, i_q} = \sum_{i_1, \cdots, i_{p-1}, i_{q+1}, \cdots, i_N}^{I_1, \cdots, I_{p-1}, I_{q+1}, \cdots, I_N} t_{i_1, \cdots, i_{p-1}, i_p, \cdots, i_q, i_{q+1}, \cdots, i_N}$.*

3.4. Tensor Construction and Fusion

To construct the base tensor, important relevant factors that influence the recommended content need to be selected as components of each order of the tensor, as shown in Table 2. In this regard, the cognitive levels are divided into low, medium, and high levels.

Table 2. Components of each order of the base tensor.

Tensor	Components
Student	student ID, stage assessment score, cognitive level
Knowledge concept	student ID, knowledge concept ID, knowledge concept score, knowledge concept learning time
Interaction	student ID, knowledge concept ID, student–system interaction, student–teacher interaction

The student tensor is constructed based on the student's corresponding correlations, as shown in Figure 1a. The student tensor S is a third-order tensor, consisting of the student ID, stage assessment score, and cognitive level. In the tensor S, each element S_{ucf} is of type Boolean. If $S_{ucf} = 1$, it means that the student u has a cognitive level of c and a stage assessment score of f. Similarly, some of the key attributes inherent to knowledge concepts are extracted to construct knowledge concept tensors such as knowledge concept ID, knowledge concept score, student ID, and knowledge concept learning time. As shown in Figure 1b, the knowledge concept tensor K is a fourth-order tensor, and each element K_{upet} is of Boolean type. If $K_{upet} = 1$, it means that the student u learns knowledge concept p with a knowledge concept score of e and a knowledge concept learning time of t. The interaction tensor represents the degree of student interaction with different objects. In the interaction tensor I shown in Figure 1c, the element I_{upssst} indicates that the student u learns the knowledge concept p with a student–system interaction of ss and the student–teacher interaction of st.

The student tensor and the knowledge concept tensor are independent of each other, while the interaction tensor can combine the two tensors, so to achieve multidimensional association analysis, tensor join is used to fuse the different tensors into a complete tensor. The student tensor, knowledge concept tensor, and interaction tensor are fused into an integrated student–knowledge concept tensor using tensor join, as shown in Figure 2a.
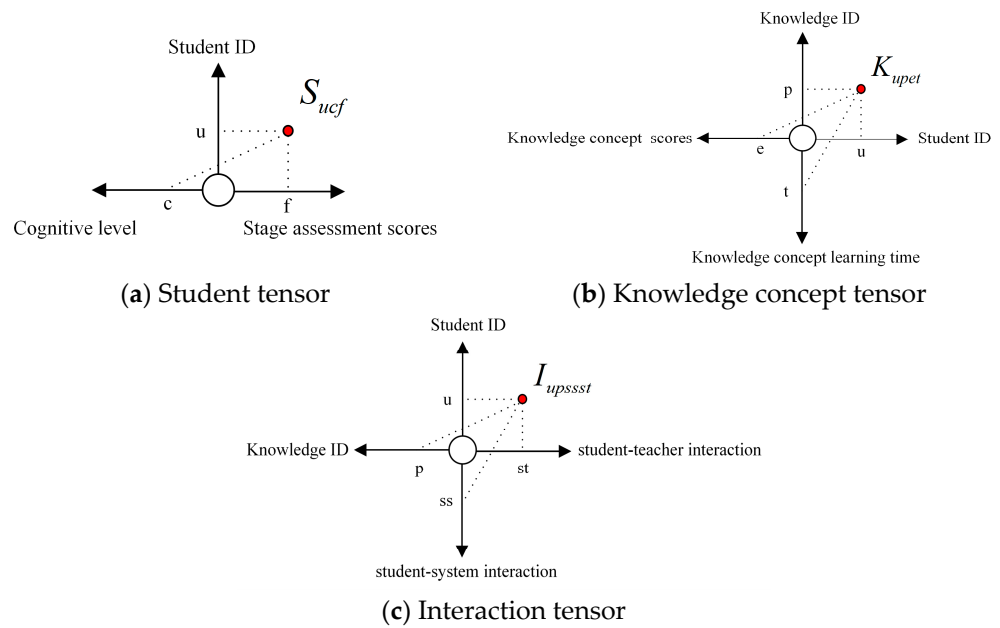
(**a**) Student tensor　　　　　　　　　　(**b**) Knowledge concept tensor

(**c**) Interaction tensor

Figure 1. Construction of the Base Tensor.
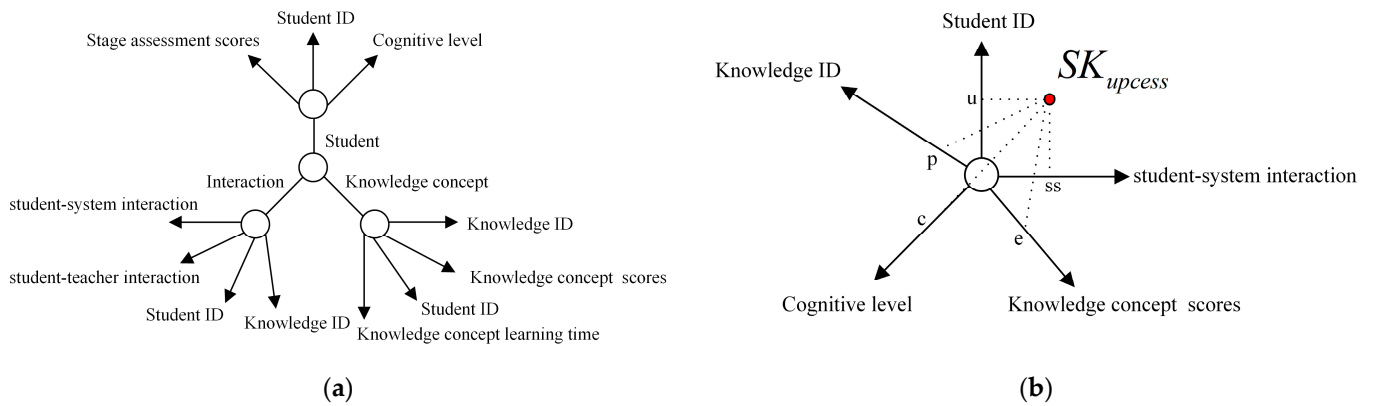

(**a**)　　　　　　　　　　　　　　　　　　　(**b**)

Figure 2. Tensor fusion and simplification. (**a**) Integrated student–knowledge concept tensor; (**b**) simplified integrated tensor.

To reduce computational costs, a tensor simplification of the integrated student–knowledge concept tensor is performed, which corresponds to the sum of all elements of the sub-tensor extracted from the original tensor along all orders except the preserved order. The simplified integrated tensor is shown in Figure 2b. The simplified fifth-order student–knowledge concept tensor SK contains the student ID, knowledge concept ID, cognitive level, knowledge concept score, and student–system interaction. The element SK_{upcess} denotes a student u with a cognitive level of c learning knowledge concept p with a knowledge concept score of e, and a student–system interaction of ss.

3.5. HOSVD

The HOSVD (High-Order Singular Value Decomposition) is a special form of the Tucker decomposition [24], which is unique and guarantees the orthogonality of the factors.

Given an Nth-order tensor $T \in R^{I_1 \times I_2 \times \cdots \times I_N}$, HOSVD consists of the following steps: First, the tensor is expanded according to $mode - n$ unfolding operation to obtain a matrix $M_{(n)}$; Second, a singular value decomposition (SVD) is performed on each expanded matrix, and the resulting left singular matrix U_n is used as the factor matrix; Third, based on the original tensor and all the factor matrices, the core tensor C is calculated, and

the approximate tensor $\widetilde{T}$ is further reconstructed. The calculation formula is given by Equation (3):

$$C = T \times_1 U_1^T \times_2 U_2^T \times_3 \cdots \times_N U_N^T$$
$$\widetilde{T} = C \times_1 U_1 \times_2 U_2 \times_3 \cdots \times_N U_N \tag{3}$$

where $\times_n (1 \leq n \leq N)$ is the mode-n product, which represents the product of the tensor and the matrix, and each U_n represents the latent eigenmatrix of the corresponding order of the original tensor T.

To minimize the loss between the approximation tensor and the original tensor, the core tensor, and the factor matrix need to be continuously updated for tensor reconstruction. To avoid overfitting, a regularization term is added after the loss function in this paper, and the loss function is given by Equation (4):

$$argmin \frac{1}{2} \|T - \widetilde{T}\|_F^2 + \frac{\lambda_1}{2} \|C\|_F^2 + \frac{\lambda_2}{2} \left(\sum_{m=0}^{N} \|U_m\|_F^2 \right) \tag{4}$$

where λ_1 and λ_2 are both greater than 0, indicating the penalty weights of the regularization term of the loss function. In this paper, the stochastic gradient descent method is used to minimize the loss function as a means of obtaining locally optimal solutions for the parameters, and the final optimal values of the parameters are obtained by iterative methods.

An example visualization of the third-order tensor HOSVD is shown in Figure 3. Each element in the reconstructed approximation tensor can be seen as a relative access likelihood of a knowledge concept. Given a particular student (or knowledge concept) and a particular condition (e.g., cognitive level), we can extract some corresponding elements (representing relative access likelihood) from the approximation tensor. Ranking them allows the generation of Top-N recommendation results for personalized recommendations.
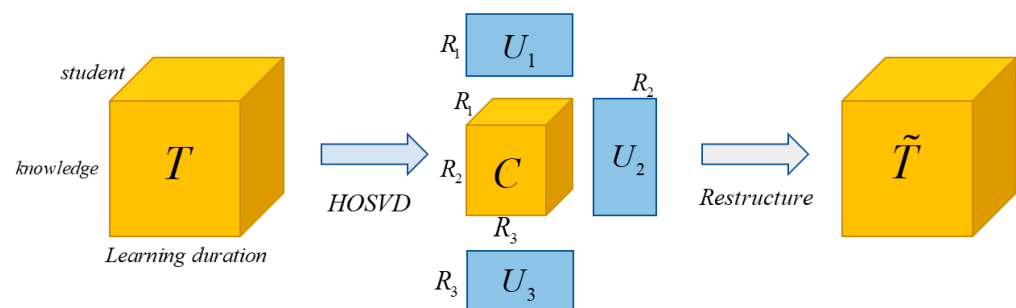

Figure 3. Visualization of third-order tensor HOSVD.

3.6. Transformer-Based Knowledge Concept Embedding

For the initial recommended sequence $RK_I = \{rk_1, rk_2, \cdots, rk_n\}$ of knowledge concepts obtained from the tensor decomposition model, this paper uses a transformer encoder layer to obtain the latent embedding of the knowledge concept sequence to capture the relationships between knowledge concepts in a high-dimensional space. This step mainly includes sequential information encoding, multi-head attention, residual connectivity, and feedforward networks [28].

3.6.1. Sequential Information Encoding

First, each knowledge concept in the knowledge concept sequence is characterized by converting it into a $d - dimensional$ embedding vector through the embedding layer. The function is given by Equation (5):

$$RK_{embedding} = Embedding(RK_I) \tag{5}$$

where $RK_I \in \mathbb{R}^n$, $RK_{embedding} \in \mathbb{R}^{n \times d}$, n is the number of knowledge concepts.

Sequential information is crucial in sequences, and to preserve the sequential relationships of knowledge concepts, sequential information is encoded in the following way:

$$
\begin{aligned}
O_{\sin}(pos, 2i) &= \sin(ord/10000^{2i/d}) \\
O_{\cos}(pos, 2i+1) &= \cos(ord/10000^{2i/d})
\end{aligned}
\tag{6}
$$

where *ord* is the position of the knowledge concept in the corresponding section of the course, *i* is used to control parity and has the same dimension as *d*. By adding the sequential information encoding to the vector of knowledge concepts at the corresponding position, a new embedding matrix $RK_O \in \mathbb{R}^{n \times d}$ can be obtained, which then contains the knowledge concept sequential information.

3.6.2. Multi-Head Attention

Multi-headed attention, which captures the characteristic relationships between knowledge concepts. After obtaining the embedding matrix RK_O, the self-attention layer performs different linear transformations on it to generate Q (Query), K (Key), and V (Value) respectively.

To calculate multi-head attention, the output of self-attention needs to be known. The output of individual self-attention can be obtained by calculating QK^T and then weighting V according to the attention matrix to obtain. The specific calculation formula is as follows:

$$
Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V
\tag{7}
$$

where $\sqrt{d_k}$ is a scaling factor that transforms the attention matrix into a normal distribution, and d_k has the same dimension as d, $Attention(Q, K, V) \in \mathbb{R}^{n \times d}$, the SoftMax function is used to normalize the weights.

Multi-head attention divides the attention matrix into multiple parts from the embedding dimension, with each part corresponding to a head. Therefore, the embedding dimension *d* must be able to divide the number of heads *h*. Each head is searched for independently and they are connected by a linear layer with the following formula:

$$
\begin{aligned}
RK_{MHA} &= concat(head_1, \ldots, head_h)W^O \\
head_i &= Attention(Q_i W_i^Q, K_i W_i^K, V_i W_i^V)
\end{aligned}
\tag{8}
$$

where $W_i^Q, W_i^K, W_i^V \in \mathbb{R}^{d/h \times d/h}$ is parameter matrices and $Q_i, K_i, V_i \in \mathbb{R}^{n \times h \times (d/h)}$.

3.6.3. Residual Connection

We add the output RK_{MHA} obtained in the previous step to the input RK_O to make the residual connection. At last, we perform Layer-Normalization on the output. The specific calculation formula is as follows:

$$
\begin{aligned}
RK_{attention} &= RK_O + RK_{MHA} \\
RK_{attention} &= LayerNorm(RK_{attention})
\end{aligned}
\tag{9}
$$

where RK_O is the knowledge concept embedding, RK_{MHA} is the output of multi-head attention.

3.6.4. Feedforward Networks

The feedforward network is a two-layer linear mapping followed by an activation function:

$$
\begin{aligned}
RK_{hidden} &= ReLu(Linear(Linear(RK_{attention}))) \\
&= \max(0, RK_{attention}W_1 + b_1)W_2 + b_2
\end{aligned}
\tag{10}
$$

where $W_1, W_2 \in \mathbb{R}^{d \times d}$ is the weight parameter of the linear layer, b_1 and b_2 are the bias parameters, and *ReLu* is the activation function.

4. TTRKRec Model

Based on the above definition, this paper proposes a knowledge concept recommendation model based on tensor decomposition and transformer reordering (TTRKRec), and the overall system architecture diagram is shown in Figure 4. The proposed model consists of the following steps: first, a student tensor (student ID, cognitive level and stage assessment score), a knowledge concept tensor (student ID, knowledge concept ID, knowledge concept score and knowledge concept learning time) and an interaction tensor (student ID, knowledge concept ID, teacher–student interaction and system interaction) containing multiple types of objects and relationships are constructed using heterogeneous data obtained from the online learning platform; secondly, a simplified integration tensor is generated by fusing and simplifying the constructed tensor, and to obtain the initial recommendation sequence of knowledge points and the student feature matrix by a higher-order singular value decomposition method based on the tensor; thirdly, the initial recommended sequence of knowledge concepts is used as input, and the transformer coding layer is used to fuse the knowledge concept sequential information and the student feature matrix to reorder the knowledge concept recommendation sequence; fourthly, a Top-N knowledge concept recommendation list was developed for each target student based on the analysis of the N knowledge concepts with the highest recommendation scores.
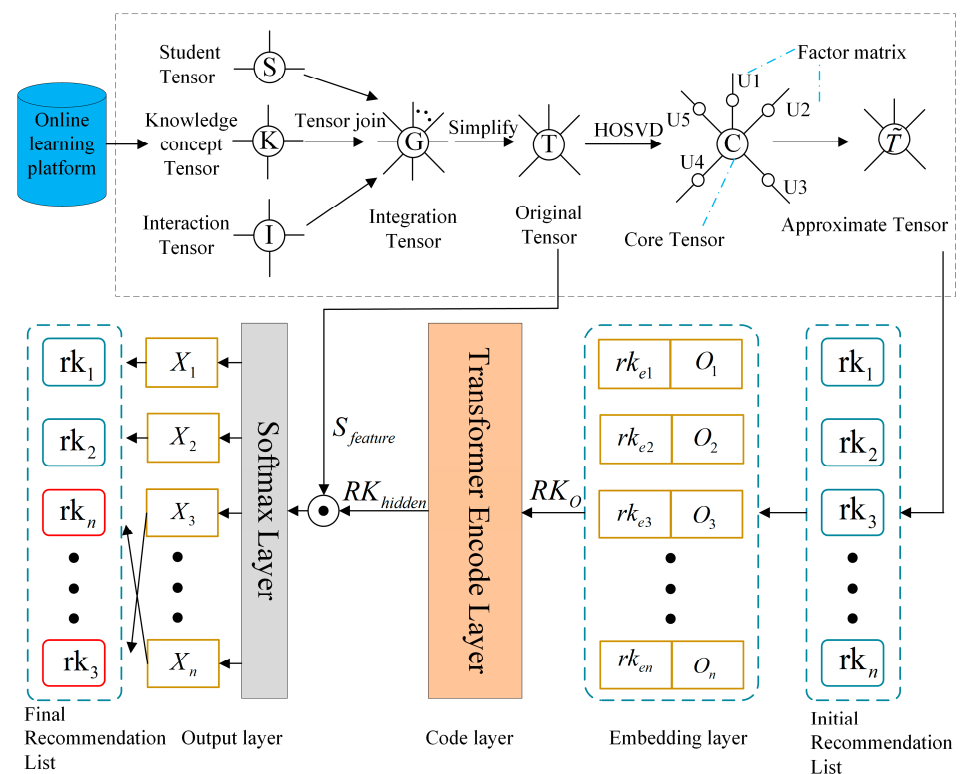

Figure 4. The overall architecture of the TTRKRec model.

4.1. Recommendations Based on Tensor Decomposition

The recommendation process based on tensor decomposition is shown in the upper part of Figure 4. The steps are as follows: first, after tensor modeling of the multidimensional time series data obtained from the online learning platform, the student tensor, the knowledge concept tensor, and the interaction tensor are obtained respectively, and then the three tensors are associated through tensor join and tensor simplification to generate the final original student–knowledge concept tensor SK; second, the model first expands the input original student–knowledge concept tensor SK according to the mode-n unfolding operation and then calculates the factor matrix U_n at each order; third, after truncating some minimal singular values according to the truncation ratio ε and extracting the important

implicit features, the core tensor C is calculated from the original tensor SK and all the factor matrices $[U_1, U_2, \ldots, U_n]$, and a less lossy approximate tensor $\widetilde{SK}$ is reconstructed by iteration; fourth, a sub-tensor $\widetilde{sk}$ is extracted from the constructed approximation tensor $\widetilde{SK}$ by fixing a specific student u and the corresponding cognitive level c; fifth, the knowledge concept access possibilities in various contexts are accumulated, in other words, the elements under all knowledge concept attribute in the $\widetilde{sk}$ are added up. The knowledge concepts are then sorted according to the magnitude of the final element values, and the final Top-N knowledge concept recommendation list is generated, as illustrated in Figure 5.
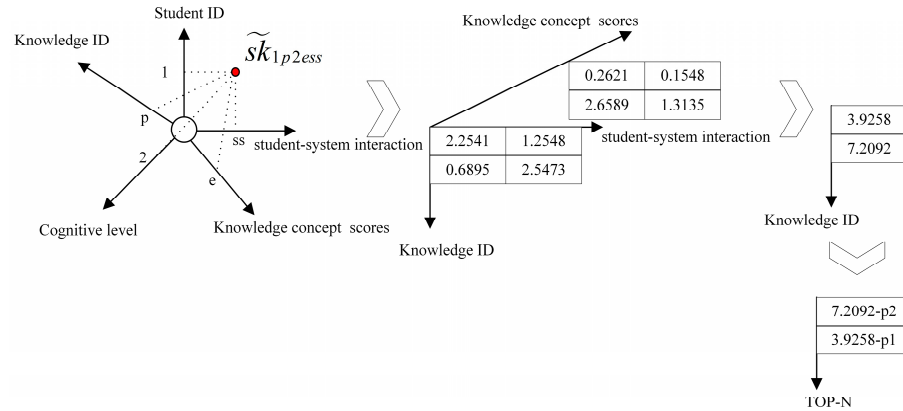

Figure 5. Examples of recommended knowledge concepts.

4.2. Reordering Based on Transformer

The reordering process based on the transformer is shown in the lower part of Figure 4. Firstly, the reordering model takes the initial recommended sequence of knowledge concepts and the knowledge concept learning sequential information as embedding information and obtains the knowledge concept latent embedding matrix RK_{hidden} after coding layer processing; secondly, the original tensor SK is tensor expanded along the student order to obtain the student feature matrix $S_{feature}$, and then the left singular value matrix is obtained by using singular value decomposition to intercept the smaller singular values and the corresponding features, to obtain the student feature matrix $S_{feature}$ which retains the key features; finally, the probability of students learning each knowledge concept, denoted as X_p, is then calculated from the SoftMax layer by fusing the features of both RK_{hidden} and $S_{feature}$ through the weight matrix W_f, which is used to rank the knowledge concepts. The specific calculation formula is as follows:

$$X_p = soft\max(RK_{hidden} \odot W_f S_{feature}{}^T) \tag{11}$$

where $W_f \in \mathbb{R}^{d \times m}$, $RK_{hidden} \in \mathbb{R}^{n \times d}$, and $S_{feature} \in \mathbb{R}^{n \times m}$.

The output layer targets the true learning label x_p of the knowledge concept and the model is trained by a loss function with the following formula:

$$L = -\frac{1}{|n|}\sum_p -[x_p \cdot \log(X_p) + (1 - x_p) \cdot \log(1 - X_p)] \tag{12}$$

where $x_p = 1$ in the case of a correct recommendation, otherwise $x_p = 0$.

5. Experiment

5.1. Experimental Dataset

In this paper, experiments were conducted on the MOOCCube dataset (Available online: http://moocdata.cn/data/MOOCCube, accessed on 8 March 2022) and Online dataset.

The MOOCCube dataset collects data from real teaching environments and consists of three main dimensions: course resources, knowledge concepts, and student behavior

records [29]. The raw data contain 706 MOOC courses, 38,181 videos, 114,563 knowledge concepts, and 199,199 MOOC students. To facilitate the tracking of students' learning processes, 312 students who took the MOOC course "Data Structures and Algorithms" and their learning behavior data were selected as the preprocessed data set. For label preprocessing, MOOCCube recorded the duration, number of times, and starting and ending times of the videos watched by the students, and removed the learning records of those who watched the videos for less than one minute.

The Online dataset consists of historical behavioral data from students who took the Data Structures and Algorithms course in 2021. It includes 207 knowledge concept videos, 39,092 video viewing records from 182 students, 4235 student–teacher interaction text records, 736 student interaction text records, and knowledge concept test data. As part of the experimental preprocessing, it was set that students who watched more than 60 s of each video were considered to have effectively learned the knowledge concept, so records of students who watched less than 60 s of each video were removed, and records of students who watched less than 1/3 of all videos were also removed, along with the corresponding student–teacher interaction and student–student interaction records. After data preprocessing, 23,043 video viewing records, 3503 student–teacher interaction text records, and 571 student interaction text records for 124 students were retained.

As the MOOCCube dataset does not contain student interaction information, a fourth-order tensor consisting of student ID, knowledge concept ID, knowledge concept learning time, and knowledge concept learning frequency was used as input to the TTRKRec model. In the MOOCCube dataset, we use sequential information formed by sequential decision relations of knowledge concepts, e.g., "K_Queue—K_Hierarchy traversal", "K_Array—K_Queue", "K_Array—K_Hierarchical traversal" can form "K_Array—K_Queue—K_Hierarchical traversal", and so on to form specific sequential information. The last 10 learning records of each student were selected as the test set according to the learning order of the students and the rest was divided into a training set (80%) and a validation set (20%).

5.2. Evaluation and Baselines

For evaluation, we used three evaluation metrics that are commonly used in recommender systems, including the area under the ROC curve (AUC), Normalized Discounted Cumulative Gain of top-K items (NDCG@K), and Mean Reciprocal Rank (MRR). In the experiments, we set K to 5 and 10. In this paper, five baseline models were used on the same dataset for comparison with the TTRKRec model, as shown in Table 3.

Table 3. Baseline descriptions.

Baselines	Description
PMF	This is a classical matrix decomposition model with a probability distribution. For knowledge concept recommendations, the method decomposes the student knowledge concept rating matrix and makes recommendations based on predicted scores [30].
ACKRec	This is a graph convolutional neural network model with an attention mechanism that transforms data into several adjacency matrices and feeds them into the model to generate embeddings of different entities [11].
Multi-HIN	This is a knowledge concept recommendation model based on a multifaceted heterogeneous information network that can naturally use rich heterogeneous context-aided information for dynamic node identification and can effectively discover and aggregate student interests [17].
FedSeqRec	This is a new horizontal federation framework for sequential recommendations that use low-rank tensor projections to model users' long-term preferences [31].
ITCA-PR	This is a tensor decomposition-based learning resource recommendation method that can recommend personalized learning resources in different contexts [24].

5.3. Implementation Details

The model proposed in this study was implemented in the PyTorch and Tensorly frameworks. In the experiments, the learning rate of the tensor recommendation model is set to 0.5, and the regularization parameter $\lambda_1 = \lambda_2 = 0.001$, the truncation ratio $\varepsilon = 0.3$, and the number of iterations of stochastic gradient descent are set to 50. the transformer reordering model sets the initial learning rate to 0.001, the weight decay to 0.00001, and the Adam optimizer is used to achieve dynamic changes in the learning rate, the number of heads in the multi-head is set to 8, and the batch size of the model is set to 60. In addition, the embedding dimension d of the transformer affects the learning representation of the sequence of knowledge concepts, and to determine the appropriate embedding dimension, a comparison of the recommended performance of the model under different embedding dimensions is carried out in this paper, as shown in Figure 6. As can be seen from the figure, the best performance is achieved when the embedding dimension d is 120, so we set the model embedding dimension $d = 120$.
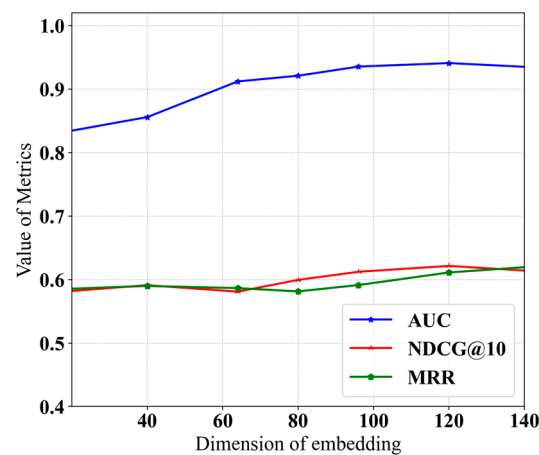

Figure 6. Impact of embedding dimension on model performance.

5.4. Experimental Results

The ROC curves for each model on the MOOCCube dataset and the Online dataset are shown in Figure 7. Comparing the area under the ROC curves shows that the TTRKRec model in Figure 7a has better AUC values than most of the models, and the TTRKRec model in Figure 7b has the best AUC values.
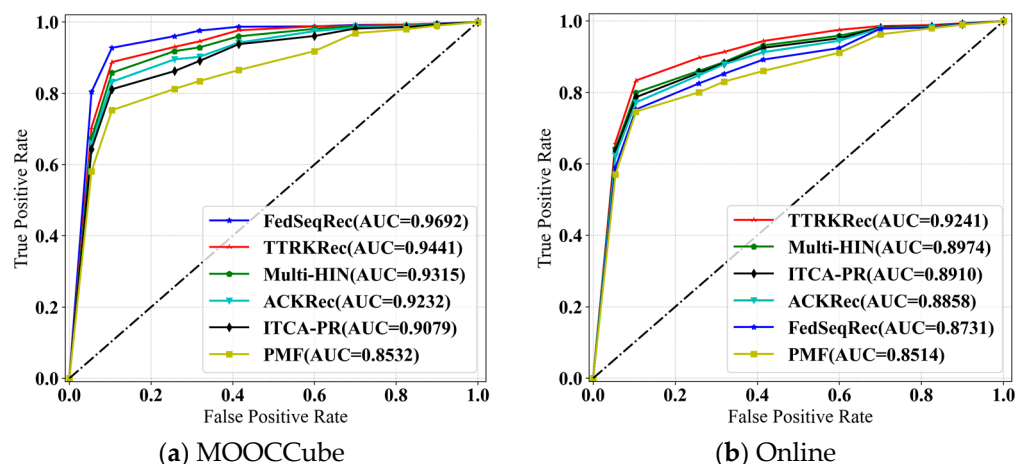

(**a**) MOOCCube   (**b**) Online

Figure 7. Comparison of the ROC curves of all models on MOOCCube and Online datasets.

Table 4 shows the results of the TTRKRec model compared to other baselines on the MOOCCube and Online datasets. The specific analysis is as follows:

Table 4. Recommendation performance of all methods on the MOOCCube and Online dataset.

Dataset	Model	AUC	NDCG@5	NDCG@10	MRR
MOOCCube	PMF	0.8532	0.2584	0.2908	0.2562
	ACKRec	0.9232	0.4635	0.5170	0.4352
	FedSeqRec	0.9692	0.3472	0.3984	0.3294
	Multi-HIN	0.9315	0.4182	0.5130	0.4140
	ITCA-PR	0.9079	0.4053	0.4584	0.4028
	TTRKRec	0.9441	0.5011	0.5715	0.4512
Online	PMF	0.8514	0.2923	0.3318	0.2912
	ACKRec	0.8858	0.3820	0.4015	0.3511
	FedSeqRec	0.8731	0.3515	0.3884	0.4028
	Multi-HIN	0.8974	0.4255	0.4697	0.4291
	ITCA-PR	0.8910	0.4212	0.4654	0.4021
	TTRKRec	0.9241	0.4862	0.5113	0.4315

(1) The proposed approach in this paper achieves the best performance in most of the evaluation metrics, which demonstrates the ability to utilize a tensor decomposition-based knowledge concept recommendation approach, and the transformer reordering solution's validity;

(2) TTRKRec has significant improvement compared to PMF based on matrix decomposition techniques. This result demonstrates the importance of tensor modeling heterogeneous data and considering multivariate data correlation to obtain entity features;

(3) TTRKRec outperforms ACKRec and FedSeqRec by considering multiple neighbors of each node and demonstrates that tensor-based heterogeneous data representation methods can capture multivariate heterogeneous information of entities more effectively;

(4) TTRKRec is more effective than Multi-HIN and ITCA-PR models that do not consider knowledge concept sequential information to perform better, showing the importance of modeling knowledge concept sequential information from multiple perspectives;

(5) The distribution and data volume of the two datasets are different, and comparing the two tables shows that the tensor recommendation model performs better in terms of the stability of recommendations;

(6) The AUC values of TTRKRec on the MOOCCube dataset were not as good as those of FedSeqRec, and it was found that the reason for this was that FedSeqRec is better at modeling students' long-term preferences in a dataset with smoother data distribution.

5.5. Ablation Studies

To illustrate the effect of different components on TTRKRec, ablation studies were conducted on two separate datasets. The experimental results are shown in Figure 8, where TTRKRec-ot is a method for disabling sequential knowledge concept information, in order to assess the importance of sequential knowledge concept information. TTRKRec-or is the method for disabling transformer reordering, in order to demonstrate that transformer reordering can improve the tensor recommendation model's recommendation effect.

Figure 8 shows the recommended performance of different variants of TTRKRec on the MOOCCube and Online datasets. From Figure 8a it can be found that on the MOOCCube dataset, TTRKRec is 9.58% and 13.27% larger than the NDCG@5 of TTRKRec-or and TTRKRec-ot, respectively. From Figure 8b, it can be found that on the Online dataset, TTRKRec is 6.5% and 17.2% larger than the NDCG@5 of TTRKRec-or and TTRKRec-ot, respectively. Combining the experimental results of the two subplots shows that TTRKRec outperforms TTRKRec-ot and TTRKRec-or in all metrics, which fully demonstrates the importance of sequential information of knowledge concepts and the effectiveness of using the transformer to do sequential information embedding model. In addition, the performance of TTRKRec-ot is worse than TTRKRec-or, suggesting that the performance of recommendations for reordering directly with the transformer is not as good as before the

reordering, further demonstrating the importance of embedding sequential information and the effectiveness of transformer as an embedding model.
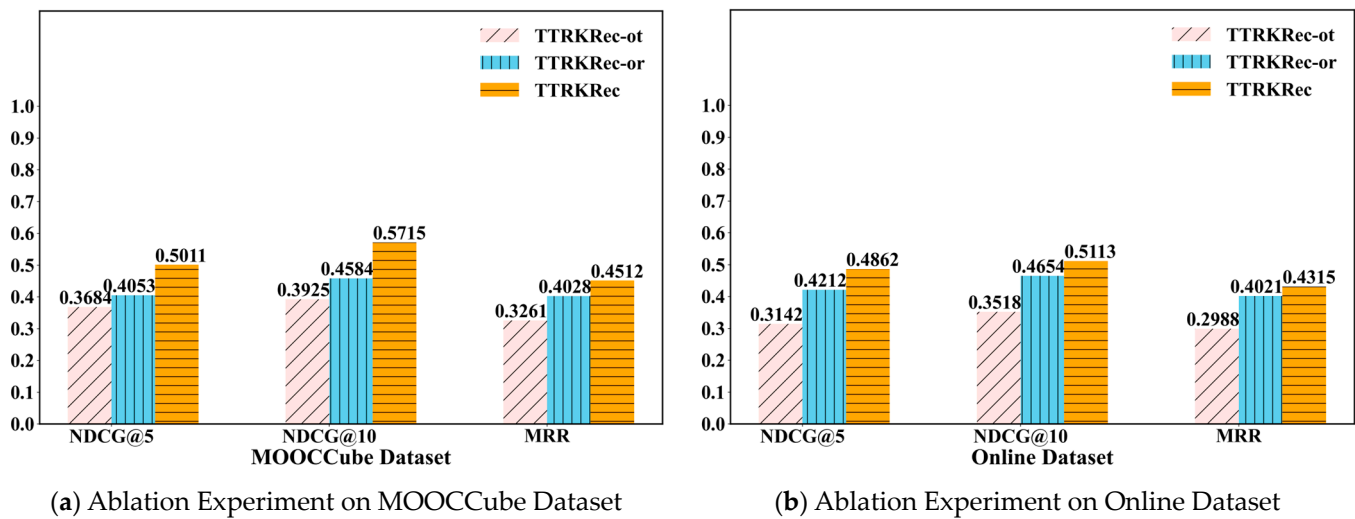

(**a**) Ablation Experiment on MOOCCube Dataset

(**b**) Ablation Experiment on Online Dataset

Figure 8. Recommended performance of different variants of TTRKRec on MOOCCube and Online datasets.

5.6. Case Study

In this section, the Top 10 knowledge concept recommendation list of a randomly selected student (Id: U_6325216) from the experimental results of the MOOCCube dataset is analyzed. Table 5 shows the recommendation lists generated by this student under different recommendation models as well as the actual learning records.

Table 5. List of recommendations under different recommendation models for student U_6325216.

Recommend List				Real Learning Record
TTRKRec	**Multi-HIN**	**ACKRec**	**ITCA-PR**	
LinkList	Order List	Data object	Substring	Top
Top	**Queue**	**Last-in first-out**	Topological sequences	Last-in First-out
Bottom	Adjacency table	Rear	**Top**	LinkList
Last-in First-out	**Top**	**Array**	Full binary tree	Search
Queue	**Binary tree**	Sequential strings	First-out First-in	Top
Binary Trees	**Graph traversal**	**Binary tree**	**Queue**	Queue
Sequential storage	Postorder traversal	**Queue**	Hash functions	Binary tree
tree	Hash Tables	**Inorder traversal**	**Search**	Array
Graph traversal	**Array**	Hash Tables	Sort	Graph traversal
Preorder traversal	Sort	Efficiency	**Graph traversal**	Inorder traversal

As can be seen from Table 5, the recommendation lists generated by the TTRKRec model are more consistent with the actual needs of students in terms of recommendation order and accuracy and can capture students' interest in knowledge concepts more accurately. This is because the TTRKRec model takes into account the sequential information of knowledge concepts and the integrity of heterogeneous information.

5.7. Discussion

Our proposed knowledge concept recommendation model can be used as a supplementary teaching module in online learning platforms to help teachers improve the design of the teaching process and meet students' individual learning needs. By comparing the recommended sequences of knowledge concepts for different students, teachers can gain insight into students' interests and learning dynamics and then adjust the course design.

Students can find their own learning pace based on the recommended knowledge concepts and improve their learning efficiency.

The above experimental results show that knowledge concept sequential information is important for knowledge concept recommendation. Similarly, corresponding information also exists in other domains, for example, in the social network domain, user relationship information is more important for community friend recommendation [32]; in the e-commerce domain, the sequential information of user visits is also important for product recommendation [33]. Therefore, it is possible to consider applying this model to social networks to recommend suitable social friends for users; similarly, it can be applied to e-commerce to recommend products that meet consumers' needs.

6. Conclusions

Course knowledge concept recommendation for online learning platforms is beneficial for promoting the effectiveness of student learning and enhancing student learning outcomes. Considering the integrity of heterogeneous information data and the importance of knowledge concept learning order to knowledge concept recommendation, this paper proposes a knowledge concept recommendation model based on tensor decomposition and transformer reordering (TTRKRec).

The TTRKRec model uses tensor modeling as a representation of heterogeneous information data to ensure data integrity, which makes data association analysis and intrinsic relationship mining more effective. More importantly, the TTRKRec model uses the transformer to complete the latent embedding of the knowledge concept sequential information in the knowledge concept recommendation sequence and the fusion of students' characteristics, further enhancing the accuracy of the knowledge concept recommendation. Experiments on two preprocessed real data sets showed that the model in this paper demonstrates better recommendation performance under different conditions than several baseline models. In future research, the placement of the model on an online learning platform to optimize the model through real student feedback will be considered. In addition, to improve students' learning effectiveness in multiple respects, correlation analysis of different recommended contents will also be considered in order to achieve multiple accurate recommendations.

Author Contributions: Conceptualization, Z.S. and Y.C.; methodology, Y.C.; software, Y.C.; validation, H.W., J.M. and H.Z.; formal analysis, H.W.; investigation, J.L.; resources, Z.S.; data curation, Y.C.; writing—original draft preparation, Y.C.; writing—review and editing, Z.S.; visualization, Y.C.; supervision, Z.S.; project administration, Z.S.; funding acquisition, Z.S. All authors have read and agreed to the published version of the manuscript.

Data Availability Statement: Online dataset is not available due to privacy concerns and MOOCCube data used can be accessed at http://moocdata.cn/data/MOOCCube (accessed on 1 February 2023).

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Wu, B.; Wang, Y. Formation mechanism of popular courses on MOOC platforms: A configurational approach. *Comput. Educ.* **2022**, *191*, 104629. [CrossRef]
2. Uddin, I.; Imran, A.S.; Muhammad, K.; Fayyaz, N.; Sajjad, M. A Systematic Mapping Review on MOOC Recommender Systems. *IEEE Access* **2021**, *9*, 118379–118405. [CrossRef]
3. Lin, L. Learning information recommendation based on text vector model and support vector machine. *J. Intell. Fuzzy Syst.* **2021**, *40*, 2445–2455. [CrossRef]
4. Li, Y.; Li, Q.; Meng, S.; Hou, J. *Transformer-Based Rating-Aware Sequential Recommendation*; Springer International Publishing: Cham, Switzerland, 2022; pp. 759–774; ISBN 0302-9743.

5. Jena, K.K.; Bhoi, S.K.; Malik, T.K.; Sahoo, K.S.; Jhanjhi, N.Z.; Bhatia, S.; Amsaad, F. E-Learning Course Recommender System Using Collaborative Filtering Models. *Electronics* **2022**, *12*, 157. [CrossRef]
6. Shen, D.; Jiang, Z. Online Teaching Course Recommendation Based on Autoencoder. *Math. Probl. Eng.* **2022**, *2022*, 8549563. [CrossRef]
7. Zhu, Y.; Lu, H.; Qiu, P.; Shi, K.; Chambua, J.; Niu, Z. Heterogeneous teaching evaluation network based offline course recommendation with graph learning and tensor factorization. *Neurocomputing* **2020**, *415*, 84–95. [CrossRef]
8. Pu, Y.; Chen, H. Exercise Recommendation Model Based on Cognitive Level and Educational Big Data Mining. *J. Funct. Space* **2022**, *2022*, 3845419. [CrossRef]
9. Diao, X.; Zeng, Q.; Li, L.; Duan, H.; Zhao, H.; Song, Z. Personalized learning path recommendation based on weak concept mining. *Mob. Inf. Syst.* **2022**, *2022*, 2944268. [CrossRef]
10. Liu, Y.; Zhang, Y.; Zhang, G. Learning path recommendation based on Transformer reordering. In Proceedings of the 2020 5th International Conference on Information Science, Computer Technology and Transportation (ISCTT), Shenyang, China, 13–15 November 2020; pp. 101–104.
11. Gong, J.; Wang, S.; Wang, J.; Feng, W.; Peng, H.; Tang, J.; Yu, P. Attentional Graph Convolutional Networks for Knowledge Concept Recommendation in MOOCs in a Heterogeneous View. In Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual, 25–30 July 2020; pp. 79–88.
12. Zhao, Z.; Yang, Y.; Li, C.; Nie, L. GuessUNeed: Recommending courses via neural attention network and course prerequisite relation embeddings. *ACM Trans. Multimed. Comput. Commun. Appl. TOMM* **2020**, *16*, 1–17. [CrossRef]
13. Shi, D.; Wang, T.; Xing, H.; Xu, H. A learning path recommendation model based on a multidimensional knowledge graph framework for e-learning. *Knowl. Based Syst.* **2020**, *195*, 105618. [CrossRef]
14. Wang, F.; Zhang, L.; Chen, X.; Wang, Z.; Xu, X. A personalized self-learning system based on knowledge graph and differential evolution algorithm. *Concurr. Comput.* **2022**, *34*, e6190. [CrossRef]
15. Huang, G.; Yuan, M.; Li, C.; Wei, Y. Personalized knowledge recommendation based on knowledge graph in petroleum exploration and development. *Int. J. Pattern Recognit. Artif. Intell.* **2020**, *34*, 2059033. [CrossRef]
16. Yang, X.; Huan, Z.; Zhai, Y.; Lin, T. Research of Personalized Recommendation Technology Based on Knowledge Graphs. *Appl. Sci.* **2021**, *11*, 7104. [CrossRef]
17. Wang, X.; Jia, L.; Guo, L.; Liu, F. Multi-aspect heterogeneous information network for MOOC knowledge concept recommendation. *Appl. Intell.* **2022**, *16*, 132. [CrossRef]
18. Gong, J.; Wang, C.; Zhao, Z.; Zhang, X. Automatic Generation of Meta-Path Graph for Concept Recommendation in MOOCs. *Electronics* **2021**, *10*, 1671. [CrossRef]
19. Ye, B.; Mao, S.; Hao, P.; Chen, W.; Bai, C. Community enhanced course concept recommendation in MOOCs with multiple entities. In Proceedings of the Knowledge Science, Engineering and Management: 14th International Conference, KSEM 2021, Tokyo, Japan, 14–16 August 2021; pp. 279–293.
20. Ling, Y.; Shan, Z. *Knowledge Concept Recommender Based on Structure Enhanced Interaction Graph Neural Network*; Springer International Publishing: Cham, Switzerland, 2022; pp. 173–186; ISBN 0302-9743.
21. Liang, Z.; Mu, L.; Chen, J.; Xie, Q. Graph path fusion and reinforcement reasoning for recommendation in MOOCs. *Educ. Inf. Technol.* **2022**, *28*, 525–545. [CrossRef]
22. Sun, Z.; Zhang, X.; Li, H.; Xiao, Y.; Guo, H. Recommender systems based on tensor decomposition. *Comput. Mater. Contin.* **2020**, *66*, 621–630. [CrossRef]
23. Hong, M.; Jung, J.J. Multi-criteria tensor model consolidating spatial and temporal information for tourism recommendation. *J. Ambient Intell. Smart Environ.* **2021**, *13*, 5–19. [CrossRef]
24. Liu, H.; Ding, J.; Yang, L.T.; Guo, Y.; Wang, X.; Deng, A. Multi-dimensional correlative recommendation and adaptive clustering via incremental tensor decomposition for sustainable smart education. *IEEE Trans. Sustain. Comput.* **2019**, *5*, 389–402. [CrossRef]
25. Shou, Z.; Wen, Y.; Chen, P.; Zhang, H. Personalized Knowledge Map Recommendations based on Interactive Behavior Preferences. *Int. J. Perform. Eng.* **2021**, *17*, 36–49.
26. Zhu, H.; Liu, Y.; Tian, F.; Ni, Y.; Wu, K.; Chen, Y.; Zheng, Q. A Cross-Curriculum Video Recommendation Algorithm Based on a Video-Associated Knowledge Map. *IEEE Access* **2018**, *6*, 57562–57571. [CrossRef]
27. Liu, H.; Yang, L.T.; Chen, J.; Ye, M.; Ding, J.; Kuang, L. Multivariate Multi-Order Markov Multi-Modal Prediction with Its Applications in Network Traffic Management. *IEEE Trans. Netw. Serv. Manag.* **2019**, *16*, 828–841. [CrossRef]
28. Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A.N.; Kaiser, A.; Polosukhin, I. Attention is all you need. *arXiv* **2017**, arXiv:1706.03762.
29. Yu, J.; Luo, G.; Xiao, T.; Zhong, Q.; Wang, Y.; Feng, W.; Luo, J.; Wang, C.; Hou, L.; Li, J. MOOCCube: A large-scale data repository for NLP applications in MOOCs. In Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, Virtual, 5 July 2020; pp. 3135–3142.
30. Mnih, A.; Salakhutdinov, R.R. Probabilistic matrix factorization. *Adv. Neural Inf. Process. Syst.* **2007**, *20*, 1257–1264.
31. Li, L.; Lin, F.; Xiahou, J.; Lin, Y.; Wu, P.; Liu, Y. Federated low-rank tensor projections for sequential recommendation. *Knowl. Based Syst.* **2022**, *255*, 109483. [CrossRef]

32. Zhu, B.; Yeung, C.H.; Liem, R.P. The impact of common neighbor algorithm on individual friend choices and online social networks. *Phys. A Stat. Mech. Its Appl.* **2021**, *566*, 125670. [CrossRef]
33. Mishra, R.; Kumar, P.; Bhasker, B. A web recommendation system considering sequential information. *Decis. Support Syst.* **2015**, *75*, 1–10. [CrossRef]