information

MDPI

Article

Personalized Course Recommendation Based on Attribute-Interaction Joint Encoding and Hypergraph Reconstruction

Jun Yi [1], Xiaoqi Han [1], Wei Zhou [1,*], Shan Xiao [2] and Ming Liu [2]

[1] School of Computer Science and Engineering, Chongqing University of Science and Technology, Chongqing 401331, China; 2010027@cqust.edu.cn (J.Y.); 2022208061@cqust.edu.cn (X.H.)
[2] Institute of Big Data and Optimization, Chongqing Polytechnic University of Electronic Technology, Chongqing 401331, China; 202025006@cquet.edu.cn (S.X.); mingliu@swu.edu.cn (M.L.)
* Correspondence: 2013004@cqust.edu.cn

Abstract

Course recommendation systems based on deep learning have demonstrated powerful feature extraction capabilities in dealing with information overload in massive open online courses (MOOCs), and have become an irreplaceable mainstream method. However, the learner–course interactions are usually scarce in reality, which limits the representation power of course recommendation. In addition, the contribution of learner and course attribute information to course recommendation has not been sufficiently explored by most existing methods. To tackle these challenges, a personalized course recommendation model based on attribute-interaction joint encoding and hypergraph reconstruction (AIHR-PCRM) is proposed in this paper. Specifically, a course hypergraph reconstruction (CHR) method is designed to construct higher-order associations for each course to explore more reliable global collaboration signals. Unlike existing hypergraph constructions that directly take learners as hyperedges, CHR explicitly couples three steps, including invalid learner elimination, high-order reachability induction, and similarity-based hyperedge filtering, to substantially raise the signal-to-noise ratio of the resulting hypergraph. Based on this, a hypergraph global collaborative learning module (HGM) can alleviate the issue of data sparsity. Then, a joint encoding module (JEM) is utilized to enhance learner behavior sequence representations by simultaneously fusing hypergraph-level global signals with attribute-level local semantics. Finally, a bidirectional self-attention module (BSM) is introduced to blend the contextual information of the learner behavior sequence, and to further provide a recommendation. Experimental results on three real-world datasets revealed that the proposed model has already achieved the best recall and ndcg scores compared to those of several existing models.

Keywords: course recommendation; hypergraph reconstruction; joint encoding; collaborative learning; attention mechanisms

Academic Editor: Yun-Chia Liang

Received: 4 May 2026
Revised: 5 June 2026
Accepted: 5 June 2026
Published: 15 June 2026

1. Introduction

Massive open online courses (MOOCs), as a new human–computer interaction teaching mode, are becoming popular among learners because of their convenience and diversity [1,2]. However, similar to other Internet applications, learners face serious information overload problems on online learning platforms, resulting in low learning efficiency and high dropout rates. The reason is that most MOOC platforms lack the correct understanding of learners' individual preferences, knowledge structure, and mastery, and often fail to

provide personalized services to learners [3]. Therefore, as an efficient information filtering technology, course recommendation (CR) is naturally applied to the MOOC platform. It analyzes the interactive data of learners on the platform to understand learners' preferences for courses and make accurate recommendations [4,5].

Despite increasing attention, two fundamental challenges remain unresolved in the MOOC course recommendation literature [6,7], which we use as the central narrative thread of this paper. (i) Sparsity of learner–course interactions: An average learner only enrolls in a handful of courses out of thousands on the platform, leaving the interaction matrix extremely sparse and the learner behavior sequence very short. (ii) Under-utilization of attribute information: Attributes such as course category, chapter, learner qualification and enrollment time carry strong semantic signals about preference and pedagogical compatibility, yet existing models rarely encode them in a way that interacts with interaction-level collaborative signals.

Existing methods address each challenge partially but each line has a clear limitation. Collaborative filtering (CF) [8] handles sparsity by smoothing interactions but cannot represent higher-order and group-level course co-occurrence. Deep recommendation models [9,10] learn nonlinear features but treat sequences as plain token streams. Graph neural network (GNN)-based methods (e.g., PCGNN [11] and HGNRec [12]) propagate signals over the bipartite learner–course graph but, because graph edges are inherently pairwise, the group-level pattern where a set of courses are co-enrolled in by the same population of learners is decomposed into independent edges and the joint information is lost. Standard hypergraph-based recommenders (e.g., [13,14]) restore higher-order group structure by allowing one hyperedge to connect many nodes, but the typical construction of one-learner–one-hyperedge produces redundant and noisy hyperedges, and most of these models do not exploit attribute information.

To address the aforementioned limitations and challenges, a personalized course recommendation model based on attribute-interaction joint encoding and hypergraph reconstruction (AIHR-PCRM) is proposed in this work. Rather than stacking existing components, AIHR-PCRM is designed around a single design principle, where the two key modules are jointly beneficial: a denoised, semantically compressed hypergraph supplies global high-order collaborative signals, and a joint encoder simultaneously injects attribute-level semantics into the same per-course representation that enters the sequence encoder. Extensive experiments evaluated on three MOOC datasets provide empirical evidence of the proposed method's effectiveness, demonstrating performance and accuracy improvements that outperform state-of-the-art approaches.

The contributions of this work are summarized as follows:

(1) A course hypergraph reconstruction (CHR) method is proposed. Unlike conventional hypergraph constructions that take each learner as a hyperedge and thus suffer from redundancy and noise, CHR explicitly combines three operations of invalid-learner elimination (denoising), high-order reachability-based hyperedge induction (structural reconstruction), and similarity-threshold-based hyperedge filtering (semantic compression) into a unified procedure.

(2) A joint encoding module (JEM) is designed that, unlike prior work which treats attributes as a separate auxiliary head, fuses learner/course attribute average embeddings with CHR-derived high-order hyperedge embeddings directly inside the per-course input of the sequence encoder. JEM acts as a complementary core component that couples global hypergraph signals with attribute-level local semantics. Together with HGM, it captures global dependencies and leverages global collaborative signals.

(3) A bidirectional self-attention mechanism module (BSM) is employed to capture long-range dependencies of arbitrary distance in the learner behavior sequence, complementing the global signals from HGM with bidirectional contextual modeling.
(4) Based on the above strategies, a personalized course recommendation model based on attribute-interaction joint encoding and hypergraph reconstruction (AIHR-PCRM) framework for recommendation is proposed to further reinforce the representation quality of recommender systems with a cross-view contrastive learning module. Extensive experiments on three benchmarks demonstrate the superiority of our proposed framework over eight state-of-the-art recommendation methods.

The rest of this work is organized as follows. The preliminaries are systematically summarized in Section 2. Section 3 presents the framework AIHR-PCRM and modules. Section 4 describes the real educational datasets used for experiments and analyzes the thorough comparison results. Finally, Section 5 concludes this article.

2. Preliminaries

2.1. Problem Statement

The task of a course recommender system is to predict the next course that matches learners' preferences based on their historical sequential interactions. Given $S = \{s_1, s_2, \ldots, s_{|S|}\}$ to denote the learner set and $C = \{c_1, c_2, \ldots, c_{|C|}\}$ to denote the course set, the behavior sequence of learner $s \in S$ is expressed as $Q_s = \left\{c_1^{(s)}, c_2^{(s)}, \ldots, c_i^{(s)}, \ldots, c_n^{(s)}\right\}$, where $c_i^{(s)} \in C$ represents the interaction between course and learner s at timestep i, and n denotes the length of this sequence. Meanwhile, given $c_a = \left\{c_1^{(a)}, c_2^{(a)}, \ldots, c_{|c_a|}^{(a)}\right\}$ to denote the course $c_i^{(s)}$ attribute set and $s_a = \left\{s_1^{(a)}, s_2^{(a)}, \ldots, s_{|s_a|}^{(a)}\right\}$ to denote the learner s attribute set, the interaction matrix $A \in \mathbb{R}^{(|S|\times|C|)}$ indicates the implicit relationships between each learner in s and their learned courses. Each entry $A_{i,j}$ in A will be set as 1 if learner s_i has adopted course c_j before and $A_{i,j} = 0$ otherwise. Based on the learner–course interaction data, the embeddings obtained by a representation function would be further used to predict the actual preference of a learner over candidate courses.

2.2. GNN-Based Recommendations

Graph neural networks (GNNs), as new efficient and scalable neural networks inspired by convolutional neural networks and graph embedding ideas, can extract and represent features of data in the graph field [15,16]. Compared with traditional deep learning methods, GNNs can reflect entities and their relationships through the constructed graph model. Generally, the process of building GNN-based recommendation systems can be roughly divided into three stages. First, a GNN-based model is constructed on the recommended entities and their interrelationships. How to encode the high-order relations based on the connection types and strength of relationships between entities is a key consideration at this stage. Next, the information propagation and updating strategies for the GNN-based model need to be decided. Choosing the most appropriate method often results in better recommendation performance. Finally, the updated node (edge, subgraph) features are extracted from the GNN-based model as their corresponding entity features, and recommendations are implemented using relevant algorithms. Successful examples that have emerged in recent years include, but are not limited to, Light-GCN [17], FDGNN [18], and SiReN [19].

Although graphs can effectively depict pairwise relationships between entities in the real world, they can only represent direct explicit interactive information between users and items. A hypergraph is composed of a set of vertices and a set of hyperedges, each of which can connect multiple vertices instead of just two. Therefore, hypergraphs have

the ability to describe complex high-order relationships between vertices and can be used to model complex networks and systems with the interactions between hyperedges and the internal nodes. To fully utilize the generalization ability of hypergraphs, researchers have begun to extend hypergraphs to design recommendation systems, like XSimGCL [20], EduGraph [21], and HHCoR [22].

On the whole, AIHR-PCRM differs from prior work in three concrete aspects: (i) Hypergraph construction: HGNN [23] and MFHCR [24] use the raw interaction matrix as the incidence matrix, i.e., one hyperedge per learner, yielding hyperedges with heavy redundancy. CHR reduces the hyperedge count by denoising and filtering. (ii) Attribute integration: HHCoR and EduGraph consume attributes only at the prediction head; AIHR-PCRM fuses attributes with hyperedge embeddings at the input level of the sequence encoder. (iii) Contrastive objective: SimGCL/XSimGCL generate positives by stochastic perturbations; AIHR-PCRM defines positives across two semantically distinct views (sequence-local vs. hypergraph-global) of the same course.

2.3. Course Recommendations

Personalized course recommendation is a popular application of recommendation systems in intelligent education. Different from ordinary product recommendation, course recommendation pays more attention to learners' short-term interaction behavior and long-term preference information, so it is more challenging. To align with the learner's historical learning activity, various online course recommendation systems have been proposed from different technical perspectives. Specifically, a high-performance course recommendation model named knowledge grouping aggregation network (KGAN) was proposed, which uses the course graph to estimate learners' potential interests automatically and iteratively [25]. Most existing course recommendation methods primarily model students' interactions with courses implicitly, failing to account for the impact of students' evolving learning interests, particularly the influence of time on course selection behavior. To address these limitations, a model based on multi-relationship and time-aware interest for personalized course recommendation (MRTI-CR) was designed, which effectively integrates heterogeneous relationships and dynamic interest evolution [26]. To extract learning preferences from video modalities in a personalized manner and link them with behavior patterns, a multi-behavior multivariate contrastive learning framework (MMCL) was employed [27]. To fully characterize and utilize the relationships among courses and other associated objects, such as teachers of courses and courses' concepts, a novel personalized interactive course recommendation scheme enhanced with a heterogeneous graph (HGCR) was developed, which smoothly combines the graph neural network with an advanced deep Q-learning neural network [28]. Excellent work in this area includes, but is not limited to, QPE [29], MECF [30], ISRA [31] and so on [32]. However, the in-depth analysis of the learner's behavior sequence and the full mining of the attribute features of learners and courses have not been fully explored, so there is still a large room for improvement in personalized course recommendation.

3. Methodology

In this section, the proposed AIHR-PCRM framework is shown in Figure 1 and the corresponding details are given.

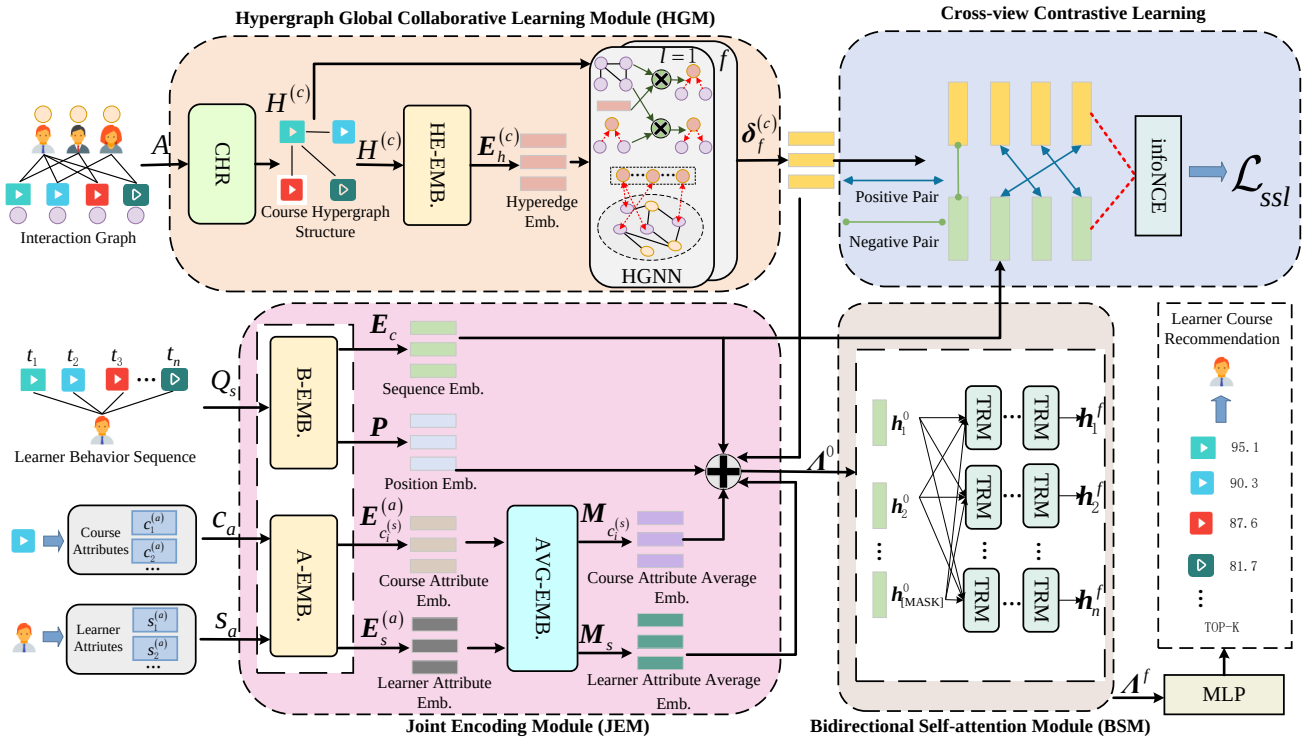

Figure 1. The architecture of the proposed course recommendation model AIHR-PCRM.

3.1. Overview

The proposed AIHR-PCRM framework roughly includes the following four components: hypergraph global collaborative learning module (HGM), joint encoding module (JEM), bidirectional self-attention module (BSM) and cross-view contrastive learning. First, learner–course interaction matrix A is sent as input to HGM. In this module, the course hypergraph correlation matrix $H^{(c)}$ is obtained through the CHR method, and then $H^{(c)}$ is sent into the hyperedge embedding layer to obtain the course hyperedge embedding matrix $\boldsymbol{E}_h^{(c)}$. Then, $H^{(c)}$ and $\boldsymbol{E}_h^{(c)}$ are sent as input to the hypergraph message-passing paradigm to obtain the course hyper-embedding matrix $\boldsymbol{\delta}_f^{(c)}$ which contains reliable high-order collaborative interaction signals between courses. In addition, learner behavior sequence Q_s, course attributes c_a, learner attributes s_a, and $\boldsymbol{\delta}_f^{(c)}$ are fed into JEM to obtain the enhanced embedding of the learner's behavior sequence $\boldsymbol{\Lambda}^0$. To further improve the perception ability of the model, BSM is employed to blend the contextual information of the learner course behavior sequence and combine it with a fully connected layer to output recommendation results. Finally, a multi-task cross-view contrastive learning strategy is utilized to obtain high-quality embeddings of courses.

3.2. Hypergraph Global Collaborative Learning Module (HGM)

In a standard learner–course bipartite-graph adjacency matrix, the fact that a learner s_1 jointly enrolls in $\{c_1, c_2, c_3\}$ is decomposed into three independent pairwise edges (s_1, c_1), (s_1, c_2), (s_1, c_3). Crucially, the group-level co-occurrence pattern where $\{c_1, c_2, c_3\}$ are jointly chosen by the same population of learners is lost. Such group-level co-occurrence often carries strong semantic signals (e.g., these three courses belong to a common learning pathway). A hyperedge can represent such a group as a single, indivisible structural unit. Moreover, in conventional GCN message-passing, the dependency among $\{c_1, c_2, c_3\}$ must be approximated via multi-hop propagation, where each additional hop introduces noise attenuation and over-smoothing. Hypergraph message-passing lets the three nodes

exchange information in a single step through a shared hyperedge, directly capturing the group dependency without multi-hop signal decay.

To alleviate the data sparsity issue of learners' behavior sequences, the CHR method is designed to construct a high-quality course hypergraph structure and utilize the hypergraph messaging mechanism to capture the implicit global collaborative relationships between courses.

3.2.1. Course Hypergraph Reconstruction (CHR)

To make CHR rigorous and reproducible, the four key terms are formalized as follows.

Definition 1 (Invalid learner). *A learner $s \in S$ is called invalid with respect to the interaction matrix A if $C(s) \cap C(s') = \emptyset$ for every other learner $s' \in S \backslash \{s\}$, where $C(s) = \left\{c \in C : A_{\{s,c\}} = 1\right\}$ is the course set enrolled by s. Intuitively, an invalid learner shares no course with anyone else and thus cannot contribute to any course-level high-order reachability.*

Definition 2 (High-order reachable course set). *Given the denoised matrix $\tilde{A}$ (after removing invalid learners), the k-th-order reachable course set of a course c is defined recursively: $R_0(c) = \{c\}$ and $R_k(c) = \{c' \in C : \exists\, s \in S$ with $c' \in C(s)$ and $C(s) \cap R_{k-1}(c) \neq \emptyset\}$. In matrix form, the closure up to order $k = 3$ is captured by $\tilde{A}\tilde{A}^{\mathrm{T}}\tilde{A}$.*

Definition 3 (Clip operator). *Let $I_{inv} \subseteq \{1, \ldots, |S|\}$ be the index set of invalid learners returned by $\Phi(\cdot)$. Then $clip(A^{\mathrm{T}}, I_{inv})$ is the column-deletion operator that removes every column of A^{T} whose index lies in I_{inv}. The resulting matrix $\tilde{A}$ has shape $|C| \times \beta$, where $\beta = |S| - |I_{inv}|$.*

Definition 4 (Similarity function). *The function $sim(\cdot, \cdot)$ is the cosine similarity between two hyperedge incidence columns: $sim(H_{\cdot,i}, H_{\cdot,j}) = \frac{H_{\cdot,i}^{T} H_{\cdot,j}}{\|H_{\cdot,i}\|_2 \|H_{\cdot,j}\|_2}$. Hyperedge pairs with similarity exceeding the threshold γ are regarded as redundant and eliminated.*

Inspired by high-order connectivity in the hypergraph-based CF, the learners' high-order reachable course sets are used to construct the course hypergraph structure. The model for constructing the course hypergraph correlation matrix is shown in Figure 2, and it can be defined as

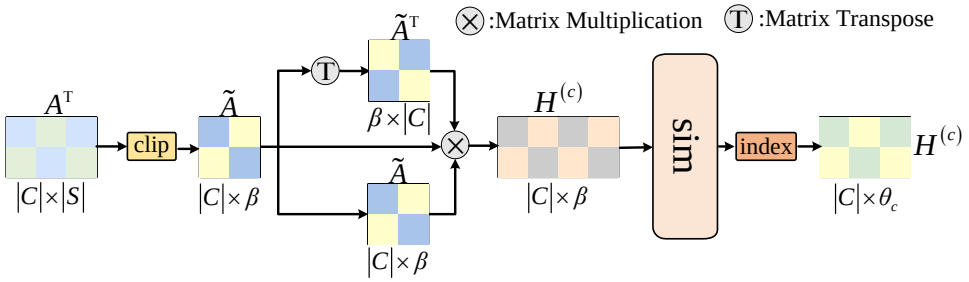

Figure 2. Course hypergraph reconstruction (CHR).

$$\tilde{A} = \mathrm{clip}\left(A^{\mathrm{T}}, \Phi\left(A^{\mathrm{T}}_{*,i}, A^{\mathrm{T}}_{*,j}\right)\right), H^{(c)} = \min\left(1, \tilde{A}\tilde{A}^{\mathrm{T}}\tilde{A}\right) \tag{1}$$

where $A^{\mathrm{T}} \in \{0,1\}^{(|C|\times|S|)}$ is the transpose of the learner–course interaction matrix, while $A^{\mathrm{T}}_{*,i}$ and $A^{\mathrm{T}}_{*,j}$ denote the i-th and j-th columns of the interaction matrix A^{T}, respectively. Function $\Phi(\cdot)$ identifies invalid learners by calculating the intersection of behavior sequences of different learners and returns the set of invalid learners' positions in the interaction matrix A^{T}, while $\tilde{A} \in \{0,1\}^{(|C|\times\beta)}$ is obtained by function $\mathrm{clip}(\cdot)$ which removes

invalid learners in the interaction matrix A^{T} based on the position information. $H^{(c)}$ is the course hypergraph correlation matrix.

The (c, s) entry of $\tilde{A}\tilde{A}^{\mathrm{T}}\tilde{A}$ counts the number of length-3 alternating paths $c \to s' \to c' \to s$ in the bipartite enrollment graph that connect course c to learner s through an intermediate learner s' and an intermediate course c'. A positive entry therefore indicates that course c is reachable from learner s through some high-order learner–course chain. The operator $\min(1, \cdot)$ is applied element-wise and converts every positive integer count to 1, yielding the $\{0,1\}$-valued hypergraph incidence matrix $H^{(c)} \in \{0,1\}^{|C|\times\beta}$. Equivalently, $\left[H^{(c)}\right]_{c,s} = \mathbf{1}\left[(\tilde{A}\tilde{A}^{\mathrm{T}}\tilde{A})_{c,s} > 0\right]$. The binarization is reasonable, which stems from the fact that within our proposed model, it is merely the existence rather than the occurrence multiplicity of high-order reachability that carries structural significance for the subsequent message-passing implemented on the hypergraph.

The hyperedge filter is formulated as follows:

$$\mu^{(c)} = \mathrm{sim}\left(H^{(c)}_{*,i}, H^{(c)}_{*,j}\right) \tag{2}$$

$$\omega^{(c)} = \mathrm{index}\left(\mu^{(c)} < \gamma\right), H^{(c)} = H^{(c)}_{*,\omega^{(c)}} \tag{3}$$

where $H^{(c)}_{*,i}$ and $H^{(c)}_{*,j}$ denote columns i and j $(i, j \in [0, \beta), i \neq j)$ in the course hypergraph correlation matrix $H^{(c)}$, respectively, while the hyperparameter γ is a threshold that controls the degree of hyperedge matching. Function $\mathrm{index}(\cdot)$ is used to identify the column indexes of the hyperedges in the course hypergraph correlation matrix that satisfy the threshold. $\mu^{(c)}$ is obtained from the matching function $\mathrm{sim}(\cdot)$, which is the set of matching values of different columns in the course hypergraph correlation matrix $H^{(c)}$, while $\omega^{(c)}$ represents the set of locations of the hyperedges in the course hypergraph correlation matrix that satisfy the inequality condition. $H^{(c)} \in \mathbb{R}^{(|C|\times\theta_c)}$ denotes the filtered course hypergraph correlation matrix, and θ_c is the number of course hyperedges.

Consider a toy example with 5 learners $\{s_1, \ldots, s_5\}$ and 4 courses $\{c_1, \ldots, c_4\}$. Suppose s_5 enrolls only in c_4, in which no other learner enrolls, then s_5 is an invalid learner and is removed. The remaining 4 learners' enrollment patterns generate high-order reachable course sets (e.g., $\{c_1, c_2, c_3\}$ appearing together), which become the candidate hyperedges. Finally, if two candidate hyperedges have nearly identical column vectors, one of them is dropped to avoid redundancy (semantic compression). The remaining hyperedges form the compact, denoised $H^{(c)}$.

To further illustrate the constructing process, the pseudo-code of CHR is shown in Algorithm 1.

Algorithm 1 Pseudocode of Courses Hypergraph Reconstruction (CHR)

Input: Transpose matrix A^{T} of learner–course interaction.
Output: Filtered course hypergraph correlation matrix $H^{(c)}$.
  Identify invalid learners in matrix A^{T} by using $\Phi(\cdot)$ function.
  Filter out invalid learners with $\mathrm{clip}(\cdot)$ function to obtain matrix $\tilde{A} \in \{0,1\}^{(|C|\times\beta)}$.
  Obtain the high-order reachable formula $\tilde{A}\tilde{A}^{\mathrm{T}}\tilde{A}$.
  Construct course hypergraph correlation matrix $H^{(c)} \in \{0,1\}^{(|C|\times\beta)}$ by Equation (1).
  Calculate the similarity $\mu^{(c)}$ for all hyperedge pairs by Equation (2).
  Introduce the hyperparameter γ as a threshold that controls the degree of hyperedge matching.
  Obtain the set of locations of the hyperedges $\omega^{(c)}$ by Equation (3).
  Obtain the filtered course hypergraph correlation matrix $H^{(c)} \in \mathbb{R}^{(|C|\times\theta_c)}$.

3.2.2. Hypergraph Message-Passing Paradigm

In the hypergraph, hyperedges are bridges that pass information during the convolution process. In this work, a hypergraph message-passing mechanism is introduced to capture the global implicit collaboration signals between courses. Before conducting hypergraph message-passing, the normalization operation of the hypergraph correlation matrix is expressed as follows:

$$\boldsymbol{\delta}^{(c)} = \overline{H^{(c)}}\boldsymbol{E}_h^{(c)}, \overline{H^{(c)}} = D_c^{-1}H^{(c)}D_e^{-1} \tag{4}$$

where $\boldsymbol{E}_h^{(c)} \in \mathbb{R}^{(\theta_c \times d)}$ denotes the course hyperedge embedding matrix which is obtained through the hyperedge embedding layer (HE-EMB), and $\overline{H^{(c)}}$ denotes the normalized course hypergraph correlation matrix. D_c and D_e are diagonal degree matrices of course nodes and course hyperedges, respectively. The normalization can be interpreted as a degree-weighted message averaging along each hyperedge. D_c^{-1} performs node-degree normalization, which prevents hub-like courses (i.e., courses appearing in many hyperedges) from dominating the propagated signal. D_e^{-1} performs hyperedge size normalization, which prevents large hyperedges (containing many courses) from contributing disproportionately. Without these factors, popular or large hyperedges would overshadow finer-grained signals and induce over-smoothing during multi-layer propagation. It is worth mentioning that in this work we improve the hypergraph normalization operation to make it more adaptable to the course recommendation work.

We refer to the hypergraph contrastive collaborative filtering (HCCF) [33] model for the hypergraph message-passing paradigm and define the hypergraph message-passing process from layer $(l-1)$ to layer (l), as shown below:

$$\boldsymbol{\delta}_l^{(c)} = \sigma\left(\overline{H^{(c)}H^{(c)}}^{\mathrm{T}}\boldsymbol{\delta}_{l-1}^{(c)}\right) \tag{5}$$

where $\sigma(\cdot)$ represents the LeakyReLU nonlinear activation function, while $\boldsymbol{\delta}_l^{(c)} \in \mathbb{R}^{(|C|\times d)}$ denotes course hyper-embeddings in the hypergraph representation space under the l-th propagation layer.

3.3. Joint Encoding Module (JEM)

The attribute information of courses and learners includes course features and learner preferences, which is a constructive guide for the performance improvement of the course recommendation system. Meanwhile, the course hyper-embedding matrix $\boldsymbol{\delta}_f^{(c)} \in \mathbb{R}^{(|C|\times d)}$ is output by the hypergraph message-passing paradigm. This matrix contains the implicit global dependencies between courses, which can effectively alleviate the issue of data sparsity of the learners' behavior sequences. To combine the above superiorities, the attribute information of courses and learners, the course hyper-embedding matrix $\boldsymbol{\delta}_f^{(c)}$, and learners' behavior sequences are joint encoded to realize the embedding enhancement of behavior sequences, which can further improve the recommendation accuracy of the AIHR-PCRM model.

3.3.1. Embedding Layer

In this layer, the course id from the learner s behavior sequence $Q_s = \{c_1^{(s)}, \ldots, c_i^{(s)}, \ldots, c_n^{(s)}\}$ is input into the behavior embedding layer to obtain its embeddings $\boldsymbol{E}_c \in \mathbb{R}^{(n\times d)}$.

$$\boldsymbol{E}_c = \left\{\boldsymbol{c}_1, \boldsymbol{c}_2, \cdots, \boldsymbol{c}_i, \cdots, \boldsymbol{c}_n\right\} \tag{6}$$

https://doi.org/10.3390/info17060598

where $\boldsymbol{c}_i \in \mathbb{R}^{(1\times d)}$ denotes the embedding of course $c_i^{(s)} \in Q_s$, and d is the embedding dimensions of course $c_i^{(s)}$.

Then, to perceive the behavior sequence order information of the learner s, a learnable positional embedding matrix $\boldsymbol{P} \in \mathbb{R}^{(n\times d)}$ is defined to obtain better course recommendation performance.

$$\boldsymbol{P} = \left\{\boldsymbol{p}_1, \boldsymbol{p}_2, \cdots, \boldsymbol{p}_i, \cdots, \boldsymbol{p}_n\right\} \tag{7}$$

where $\boldsymbol{p}_i \in \mathbb{R}^{(1\times d)}$ denotes the positional embedding of course $c_i^{(s)}$.

Eventually, the attribute information $c_a = \left\{c_1^{(a)}, c_2^{(a)}, \ldots, c_{|c_a|}^{(a)}\right\}$ of course $c_i^{(s)}$ and that $s_a = \left\{s_1^{(a)}, s_2^{(a)}, \ldots, s_{|s_a|}^{(a)}\right\}$ of learner s are input into the attribute embedding layer to obtain the attribute embeddings $\boldsymbol{E}_{c_i^{(s)}}^{(a)} \in \mathbb{R}^{(|c_a|\times d)}$ and $\boldsymbol{E}_s^{(a)} \in \mathbb{R}^{(|s_a|\times d)}$ of course $c_i^{(s)}$ and learner s, respectively.

$$\boldsymbol{E}_{c_i^{(s)}}^{(a)} = \left\{\boldsymbol{c}_1^{(a)}, \boldsymbol{c}_2^{(a)}, \cdots, \boldsymbol{c}_i^{(a)}, \cdots, \boldsymbol{c}_{|c_a|}^{(a)}\right\} \tag{8}$$

$$\boldsymbol{E}_s^{(a)} = \left\{\boldsymbol{s}_1^{(a)}, \boldsymbol{s}_2^{(a)}, \cdots, \boldsymbol{s}_i^{(a)}, \cdots, \boldsymbol{s}_{|s_a|}^{(a)}\right\} \tag{9}$$

where $\boldsymbol{c}_i^{(a)} \in \mathbb{R}^{(1\times d)}$ and $\boldsymbol{s}_i^{(a)} \in \mathbb{R}^{(1\times d)}$ denote the embedding of the i-th attribute of the course $c_i^{(s)}$ and learner s, respectively, while d is the embedding dimensions.

For implementation, attributes are encoded according to their type. (i) Categorical attributes (e.g., Course Category, Keywords, Chapter, Learner Qualification): One-hot indices are mapped through a learnable embedding layer to d-dimensional vectors. (ii) Numerical attributes (e.g., Age, Duration, Enroll Time): Values are first discretized into 10 equal-frequency bins, then embedded similarly. (iii) Text-like attributes (e.g., Course Name): Encoded by averaging pretrained word embeddings of their tokens. (iv) Missing values: For each attribute we reserve a special UNKNOW index, which avoids the bias introduced by zero-imputation in embedding-based models.

3.3.2. Joint Encoding Layer

To inject the embedding information into the training process of the proposed model, the attribute embedding matrix $\boldsymbol{E}_{c_i^{(s)}}^{(a)}$ and $\boldsymbol{E}_s^{(a)}$, the behavior sequence embeddings $\boldsymbol{E}_c$, the positional embeddings $\boldsymbol{P}$, and the course hyper-embedding matrix $\boldsymbol{\delta}_f^{(c)}$ are jointly encoded to obtain the enhanced embedding $\boldsymbol{h}_i^0 \in \mathbb{R}^{(1\times d)}$ for the i-th course in the behavior sequence of learner s. To further simplify the computation, $\boldsymbol{\Lambda}^0 \in \mathbb{R}^{(n\times d)}$ is used to represent the enhanced embeddings of all courses in the sequence. Specific definitions are given below:

$$\boldsymbol{M}_{c_i^{(s)}} = \frac{1}{|c_a|}\sum_{i=1}^{|c_a|}\boldsymbol{c}_i^{(a)}, \quad \boldsymbol{M}_s = \frac{1}{|s_a|}\sum_{i=1}^{|s_a|}\boldsymbol{s}_i^{(a)} \tag{10}$$

$$\boldsymbol{h}_i^0 = \boldsymbol{M}_{c_i^{(s)}} \oplus \boldsymbol{M}_s \oplus \boldsymbol{c}_i \oplus \boldsymbol{\delta}_{i,f}^{(c)} \oplus \boldsymbol{p}_i \tag{11}$$

where $\boldsymbol{M}_{c_i^{(s)}} \in \mathbb{R}^{(1\times d)}, \boldsymbol{M}_s \in \mathbb{R}^{(1\times d)}$ represents the attribute-averaging embeddings of course $c_i^{(s)}$ and learner s, respectively, which are obtained through attribute-averaging embedding layer (AVG-EMB). $\boldsymbol{c}_i \in \mathbb{R}^{(1\times d)}$ denotes the initialized embeddings of course $c_i^{(s)}$, while $\boldsymbol{\delta}_{i,f}^{(c)} \in \mathbb{R}^{(1\times d)}$ is the hyper-embeddings of course $c_i^{(s)}$. $\boldsymbol{p}_i \in \mathbb{R}^{(1\times d)}$ represents the learnable positional embeddings of course $c_i^{(s)}$. From Equation (11) the gradient of the

downstream loss L with respect to the attribute embedding ${c_i}^{(a)}$ is $\frac{\partial L}{\partial {c_i}^{(a)}} = \frac{\partial L}{\partial h_i^0} \cdot \frac{\partial h_i^0}{\partial M_{{c_i}^{(s)}}} \cdot \frac{\partial M_{{c_i}^{(s)}}}{\partial {c_i}^{(a)}}$, and the gradient with respect to a hyperedge embedding $e_k \in E_h^{(c)}$ is $\frac{\partial L}{\partial e_k} = \frac{\partial L}{\partial h_i^0} \cdot \frac{\partial h_i^0}{\partial \delta_{i,f}^{(c)}} \cdot \frac{\partial \delta_{i,f}^{(c)}}{\partial e_k}$. Because both gradient chains share the common factor $\partial L / \partial h_i^0$, the attribute branch and the hypergraph branch co-adapt during training: the gradient flowing through h_i^0 is jointly shaped by the two sources of information. Removing JEM cuts the first chain and removing HGM cuts the second; in either case the shared adaptation pathway is broken. This co-adaptation provides a mechanistic reason why JEM and HGM are complementary rather than independent.

3.4. Bidirectional Self-Attention Module (BSM)

In the actual course recommendation scenario, the learners' preference information for courses is also implicit in their behavior sequences. A bidirectional self-attentive encoder for course recommendation is introduced, which enables the AIHR-PCRM model to capture dependencies directly at any distance between courses in the sequence of learner behavior. Specifically, a transformer-based bidirectional self-attentive encoder is used for behavior perception, and the enhanced embeddings $\boldsymbol{\Lambda}^0 \in \mathbb{R}^{(n \times d)}$ of the behavior sequences of the learner s are utilized as the input of the encoder, while stacking multiple layers of transformer and representing the hidden embeddings that pass through the output of the l-th layer as $\boldsymbol{\Lambda}^l \in \mathbb{R}^{(n \times d)}$.

3.4.1. Multi-Head Self-Attention Mechanism

Recent research has demonstrated the superiority of transformer in modeling dependencies between different representation pairs in sequence recommendation [34]. Moreover, the self-attention mechanism in the transformer framework is the key component that performs dependency sensing and information aggregation between data points (e.g., words, courses, pixels) [35]. To enable the proposed model to perceive information from different subspaces at different locations in the learner's behavior sequence, a multi-head self-attention mechanism is applied to perform information integration from different subspaces. Specifically, firstly, the trainable matrix is utilized to linearly project $\boldsymbol{\Lambda}^l$ into m subspaces, and then the self-attention function $\mathrm{Atten}(\cdot)$ is used to compute the m-head attention embeddings $head_i \in \mathbb{R}^{(n \times d/m)}$, respectively, and immediately after that, the multi-head attention function $\mathrm{MH}(\cdot)$ is employed to output the multi-head attention results by horizontally concatenating the m-head attention embeddings and further linearly projecting them.

$$\begin{aligned} head_i &= \mathrm{Atten}\left(\boldsymbol{\Lambda}^l \boldsymbol{W}_i^Q, \boldsymbol{\Lambda}^l \boldsymbol{W}_i^K, \boldsymbol{\Lambda}^l \boldsymbol{W}_i^V\right) \\ &= \mathrm{softmax}\left(\frac{\left(\boldsymbol{\Lambda}^l \boldsymbol{W}_i^Q\right)\left(\boldsymbol{\Lambda}^l \boldsymbol{W}_i^K\right)^{\mathrm{T}}}{\sqrt{d/m}}\right)\boldsymbol{\Lambda}^l \boldsymbol{W}_i^V \end{aligned} \tag{12}$$

$$\mathrm{MH}\left(\boldsymbol{\Lambda}^l\right) = [head_1; head_2; \ldots; head_m]\boldsymbol{W}^O \tag{13}$$

where $\boldsymbol{W}_i^Q \in \mathbb{R}^{(d \times d/m)}$, $\boldsymbol{W}_i^K \in \mathbb{R}^{(d \times d/m)}$, $\boldsymbol{W}_i^V \in \mathbb{R}^{(d \times d/m)}$, and $\boldsymbol{W}^O \in \mathbb{R}^{(d \times d)}$ denote the learnable projection matrix, while $\boldsymbol{\Lambda}^l \boldsymbol{W}_i^Q \in \mathbb{R}^{(n \times d/m)}$, $\boldsymbol{\Lambda}^l \boldsymbol{W}_i^K \in \mathbb{R}^{(n \times d/m)}$, and $\boldsymbol{\Lambda}^l \boldsymbol{W}_i^V \in \mathbb{R}^{(n \times d/m)}$ represent the query, key, and value matrices in the self-attention mechanism, respectively. The temperature parameter $\sqrt{d/m}$ is introduced to avoid the phenomenon of gradient vanishing.

3.4.2. Feed-Forward Network

In this work, a feed-forward network is used to inject the nonlinear transformation into the newly generated hidden layer embedding, and the nonlinear transformation layer is denoted as follows:

$$\mathrm{CFFN}\left(\boldsymbol{\Lambda}^{l}\right) = \left[\mathrm{FFN}\left(\boldsymbol{h}_{1}^{l}\right)^{\mathrm{T}}; \cdots ; \mathrm{FFN}\left(\boldsymbol{h}_{n}^{l}\right)^{\mathrm{T}}\right]^{\mathrm{T}} \tag{14}$$

$$\mathrm{FFN}(\boldsymbol{x}) = \mathrm{GLUE}\left(\boldsymbol{x}\boldsymbol{W}_{1}^{l} + \boldsymbol{b}_{1}^{l}\right)\boldsymbol{W}_{2}^{l} + \boldsymbol{b}_{2}^{l} \tag{15}$$

In the feed-forward network, the $\mathrm{FFN}(\cdot)$ function is utilized to realize the two-layer nonlinear transformation, and the $\mathrm{GLUE}(\cdot)$ activation function is used for nonlinear activation in the middle layer. The $\mathrm{CFFN}(\cdot)$ function is employed to horizontally concatenate the hidden layer embeddings at all locations that have been injected into the nonlinear transformation. In addition, $\boldsymbol{W}_{1}^{l} \in \mathbb{R}^{(d\times k)}$, $\boldsymbol{W}_{2}^{l} \in \mathbb{R}^{(k\times d)}$, $\boldsymbol{b}_{1}^{l} \in \mathbb{R}^{(1\times k)}$, and $\boldsymbol{b}_{2}^{l} \in \mathbb{R}^{(1\times d)}$ denote the learnable projection matrices and bias terms, respectively, while k is the projection matrices' and bias terms' dimensions.

3.4.3. Stacked Multi-Layer Transformer

To further enhance the course recommendation effect of the AIHR-PCRM model, multi-layer transformers are stacked. In addition, the $\mathrm{Dropout}(\cdot)$ function is applied to prevent model overfitting, and the layer normalization $\mathrm{LN}(\cdot)$ function for network stabilization and network training acceleration are expressed as follows:

$$\boldsymbol{\Lambda}^{l} = \mathrm{Trm}\left(\boldsymbol{\Lambda}^{l-1}\right) \tag{16}$$

$$\mathrm{Trm}\left(\boldsymbol{\Lambda}^{l-1}\right) = \mathrm{LN}\left(\boldsymbol{\Gamma}^{l-1} + \mathrm{Dropout}\left(\mathrm{CFFN}\left(\boldsymbol{\Gamma}^{l-1}\right)\right)\right) \tag{17}$$

$$\boldsymbol{\Gamma}^{l-1} = \mathrm{LN}\left(\boldsymbol{\Lambda}^{l-1} + \mathrm{Dropout}\left(\mathrm{MH}\left(\boldsymbol{\Lambda}^{l-1}\right)\right)\right) \tag{18}$$

where $\mathrm{Trm}(\cdot)$ denotes one of the layers of the multi-layer transformer, while $\boldsymbol{\Gamma}^{l-1}$ denotes the hidden layer embedding in the $(l-1)$-th layer transformer after layer normalization and the multi-head attention mechanism.

3.5. Prediction

Eventually, a multi-layer perceptron layer (MLP) is employed to further learn the interaction signals between dense features and inputs the output prediction result matrix into a $\mathrm{softmax}(\cdot)$ function to convert numerical data into probabilistic data, which is formulated as follows:

$$\Psi = \mathrm{softmax}\left(\mathrm{MLP}\left(\boldsymbol{\Lambda}^{f}\right)\right), \boldsymbol{Pred}_{s} = \mathrm{EX}(\Psi) \tag{19}$$

where $\boldsymbol{\Lambda}^{f} \in \mathbb{R}^{(n\times d)}$ represents the output from the last layer of the stacked multi-layer transformer. $\Psi \in \mathbb{R}^{(n\times|C|)}$ denotes the probability estimation matrix, which is obtained through the MLP layer and the $\mathrm{softmax}(\cdot)$ function and using the $\mathrm{EX}(\cdot)$ function to obtain the last row of the probability matrix Ψ, and $Pred_{s}$ denotes the probability prediction value of the learner s who chooses any course in the course set as the next course. The Top-K scheme is used for course recommendation, where the AIHR-PCRM model recommends only the top K courses with the highest probability.

3.6. Objective Function

3.6.1. Cross-Entropy Loss

To make the proposed model consistent with the actual learner course recommendation scenarios, the cloze task is used as the training target of the AIHR-PCRM model to realize

the bidirectional information modeling of the learners' behavior sequences. Since the bidirectional self-attention module (BSM) causes output course representations to contain information about the target course, we randomly mask the courses in the learners' behavior sequences at a ratio of ρ to avoid the problem of label leakage. Then, unmasked courses are utilized to predict masked courses. The cross-entropy loss function is expressed as follows:

$$\mathcal{L}_p = \frac{1}{|B|} \sum_{t \in B, u \in U} -\log\left(\frac{\exp(\Psi_{u,t})}{\sum_{j \in C} \exp(\Psi_{u,j})}\right) \tag{20}$$

where $\Psi \in \mathbb{R}^{(n \times |C|)}$ denotes the probability estimation matrix. B denotes the true position set of masked courses in each batch of data, and U represents the set of masked positions corresponding to B. C denotes the set of all courses.

3.6.2. Cross-View Contrastive Learning

For the same course c_i, the initial embedding c_i (sequence view) is learned from each learner's local behavior context, while the hypergraph embedding $\delta_{i,f}^{(c)}$ is learned from cross-learner course co-occurrence patterns via the hypergraph message-passing paradigm (HMP). The two embeddings encode the same semantic entity from complementary perspectives from local-sequential to global-collaborative. Forcing them to be close is therefore a natural multi-view consistency constraint, encouraging both encoders to converge on a shared, discriminative representation. For each anchor course c_i in a training batch of size $B = 256$ learner sequences, the positive sample is its own hypergraph-view embedding $\delta_{i,f}^{(c)}$. Candidate negatives are formed by all other courses $c_{i'}$ ($i' \neq i$) whose hypergraph views appear in the same batch (in-batch negative sampling). After pooling, this yields on average $\sim$50 candidates per batch. A candidate $c_{i'}$ is masked out from the negative set whenever $c_{i'} \in \mathcal{N}_h(c_i)$, where $\mathcal{N}_h(c_i) = \left\{c' : \exists h \in \mathcal{E},\ c_i \text{ and } c' \text{ both connected to } h \text{ in } H^{(c)}\right\}$ denotes the hyperedge-neighbourhood of c_i. This masking rule prevents semantically related courses (i.e., those that already share a high-order co-enrollment hyperedge with the anchor) from being treated as hard negatives, which would otherwise distort the learned representation. The effective number of negatives per anchor is therefore approximately $(M-1) - |\mathcal{N}_h(c_i) \cap B|$, typically 30–40 in our experiments. Unlike SimGCL/XSimGCL which generate positives via stochastic perturbations, AIHR-PCRM defines positives across two semantically distinct views (sequence-local vs. hypergraph-global) of the same course.

The InfoNCE contrastive loss is defined as

$$\mathcal{L}_{ssl} = \sum_{i=0}^{Y} -\log\left(\frac{\exp\left(s\left(\boldsymbol{c}_i, \boldsymbol{\delta}_{i,f}^{(c)}\right)/\tau\right)}{\sum_{i'=0}^{Y} \exp\left(s\left(\boldsymbol{c}_i, \boldsymbol{\delta}_{i',f}^{(c)}\right)/\tau\right)}\right) \tag{21}$$

where Y denotes the set of unmasked course positions in each batch of data, while $s(\cdot)$ represents the similarity between the two vectors, set as a cosine similarity function, and the hyperparameter τ is a tunable temperature hyperparameter.

3.6.3. Multi-Task Training

To strengthen the course recommendation performance of the proposed model, a multi-task training strategy is used to optimize the contrast loss and cross-entropy loss, which is formulated as

$$\mathcal{L}_t = \mathcal{L}_p + \lambda_1 \mathcal{L}_{ssl} + \lambda_2 \|\Theta\|_F^2 \tag{22}$$

where λ_1 and λ_2 are hyperparameters for two loss terms as $\mathcal{L}_{ssl}$ and $\mathcal{L}_2$, respectively, while $\|\Theta\|_F^2$ denotes the $\mathcal{L}_2$ regularization term for weight decay.

3.7. Complexity Analysis

We analyze the time and memory complexity of AIHR-PCRM module by module.

(1) HGM. The dominant cost of CHR is the high-order reachability computation $\tilde{A}\tilde{A}^{\mathrm{T}}\tilde{A}$ with time complexity $O(|C|^2 \cdot \beta)$ for the learner–course interaction matrix, where $\beta = S$-invalid learners. The hyperedge filter requires pairwise cosine similarity over the β columns, with cost $O(|C| \cdot \beta^2)$ in the worst case, but is reduced to $O(|C| \cdot \beta \cdot \theta_c)$ after early termination on similarity violations. Crucially, CHR is computed only once as a preprocessing step. The per-layer HMP complexity is $O(|C| \cdot \theta_c \cdot d)$, which is comparable to LightGCN on a graph of similar density.

(2) JEM. Pure embedding lookup and averaging, with cost $O(n \cdot d + |c_a \cdot d| + |s_a \cdot d|)$ per learner, which can be negligible compared to HGM and BSM.

(3) BSM. Standard transformer cost, $O(L \cdot n^2 \cdot d)$ per training step, where L is the number of stacked transformer layers and n is the sequence length.

Overall, the total time complexity of the proposed AIHR-PCRM is $O(|C|^2 \cdot \beta + |C| \cdot \beta \cdot \theta_c + |C| \cdot \theta_c \cdot d + L \cdot n^2 \cdot d)$.

4. Experiments and Results

In this section, the experiment design and the results are shown by using the proposed course recommendation model. The AIHR-PCRM model's practicability and effectiveness are demonstrated through comparing it with several existing recommendation models. Additionally, the impact of different modules and hyperparameter settings on the performance of the proposed model is analyzed, and the robustness of our proposed model in handling sparse data is further demonstrated through a data sparsity study.

4.1. Datasets

We use three datasets: *XuetangX* [36], *MOOCCube* [37], and *CNPC*. The dataset statistics are summarized in Table 1.

Table 1. Statistical information of the datasets.

Dataset	User	Course	Learner–Course Interaction	Average Learner Interaction
XuetangX	82,535	1032	458,454	5.6
MOOCCube	55,203	706	354,541	6.4
CNPC	31,890	238	325,199	10.2

We adopt a strict per-user chronological leave-one-out split as our main experimental protocol. For each learner whose interaction sequence has length $n \geq 3$, interactions are sorted by timestamp; the most recent interaction is held out as the test sample, the second-to-most-recent as the validation sample, and the remaining $(n - 2)$ interactions form the training data. Learners with fewer than three interactions are filtered out during preprocessing because they cannot yield disjoint train/validation/test samples under leave-one-out. This filtering rule is consistent with BERT4Rec and SASRec, and removes fewer than 1.5% of learners on each of the three datasets (final retained learners: 81,604/54,887/31,612 for XuetangX/MOOCCube/CNPC). As a complementary robustness check, we additionally evaluate all methods under a strict global temporal split in Appendix B; the relative ranking of all methods is preserved.

The attribute information of learners and courses, including learner preferences and course features, can improve the course recommendation accuracy of the proposed model. Therefore, the attribute information for courses and learners is collected from three real

datasets, as shown in Table 2. All three datasets (XuetangX, MOOCCube, CNPC) are publicly released with prior anonymization of learner identifiers, and our use complies with their respective data-use agreements. In response to reader concerns about fairness and privacy, we removed the sensitive attributes Gender and City from the input of AIHR-PCRM in the experiments. The remaining attributes are academically grounded (e.g., Qualification, Enroll Time, Age Binned).

Table 2. Statistics of attribute information.

	Attributes
Course	Name, Categories, Keywords, Chapter, Duration, ...
Learner	Name (anonymized), Age, Qualification, Enroll_time, ...

4.2. Baseline Methods

To evaluate our proposed method, we compare it with the following baselines:

HGCR [28]: This is a novel deep reinforcement learning-based personalized interactive course recommendation scheme enhanced with the heterogeneous graph, which smoothly combines the graph neural network with an advanced deep Q-learning neural network.

MECF [30]: To cope with the challenge of using a single modality or a limited subset of modalities for recommendation, a multimodal enhanced online learning collaborative filtering method is designed.

Light-GCN [17]: This method learns the student and course embeddings by linearly spreading the student and course embeddings on the student–course interaction graph, and uses the weighted sum of the embeddings learned in all layers as the final embedding.

ISRA [31]: The proposed approach uses the advantages of personalized recommendation algorithms in filtering applications to reconstruct the music performance training system.

HGNN [23]: The model treats learners as hyperedges of the set of courses in a hypergraph and transforms the task of learning learners' representations into an embedding that induces hyperedges, and subsequently, a hyperedge-based graph attention network is designed based on it.

MFHCR [24]: This model utilizes a hypergraph to model the interactive information between users and courses to obtain a comprehensive user representation to generate recommendation results.

XSimGCL [20]: Based on SimGCL, this method is improved by using noise enhancement and inter-layer contrastive learning, and achieves the same effect as SimGCL by unifying task recommendation and task comparison, while reducing the time complexity of the model.

HHCoR [22]: It constructs an online course hypergraph as the environment to capture the complex relationships and historical information by considering all entities, and designing a multi-channel propagation mechanism to aggregate embeddings in the online course hypergraph and extract user interest through an attention layer.

4.3. Evaluation Metrics

To measure the accuracy of the proposed model, we use two Top-N evaluation metrics, including Recall@N and NDCG@N, while R and N are used uniformly as simplified expressions for the evaluation metrics Recall and NDCG. To improve the reliability of the experimental results, N is set to 10, 20.

4.4. Parameter Settings

To compare fairly, all models in our experiments are trained from scratch and are initialized with the Xavier method. We use Adam optimizer with the learning rate of 1×10^{-3} and 0.96 decay ratio for model inference. The embedding dimension for the proposed model is set to 128. In hypergraph learning, the number of hyperedges to filter is chosen from $\{0.5, 0.6, 0.7\}$. Moreover, the number of layers in the hypergraph message-passing architecture is set to two. The hyperparameters λ_1 and λ_2 are chosen from $\{1 \times 10^{-5}, 1 \times 10^{-4}, 1 \times 10^{-3}, 1 \times 10^{-2}\}$, and the temperature parameter τ is chosen from $\{0.3, 0.5, 1, 5, 10\}$.

For complete reproducibility, we report all hyperparameter settings in Table 3. All reported numbers in Tables 4 and 5 are the mean $\pm$ standard deviation over five independent runs with random seeds $\{42, 123, 456, 789, 1000\}$. For each dataset and each metric we additionally performed a paired two-tailed t-test between AIHR-PCRM and the strongest baseline (HHCoR). All improvements in Table 4 are statistically significant at the 1% level ($p < 0.01$), with most significant at the 0.1% level. Wilcoxon signed-rank tests give identical conclusions.

Table 3. Full hyperparameter settings used in all experiments.

Hyperparameter	**XuetangX**	**MOOCCube**	**CNPC**
Optimizer	Adam	Adam	Adam
Learning rate (decay)	1×10^{-3} (0.96)	1×10^{-3} (0.96)	1×10^{-3} (0.96)
Embedding dimension d	128	128	128
HMP layers	2	2	2
Transformer layers L	2	2	2
Attention heads m	4	4	4
Head dimension d/m	32	32	32
FFN hidden dim k	256	256	256
Dropout probability	0.1, 0.3, 0.6, 0.8	0.1, 0.3, 0.6, 0.8	0.1, 0.3, 0.6, 0.8
Batch size	256	256	256
Max sequence length n	128	128	128
Mask ratio ρ	0.2, 0.4, 0.6	0.2, 0.4, 0.6	0.2, 0.4, 0.6
Max epochs	200	200	200
Early-stop patience	20 (on val R@20)	20 (on val R@20)	20 (on val R@20)
Random seeds	42, 123, 456, 789, 1000	42, 123, 456, 789, 1000	42, 123, 456, 789, 1000
Hyperedge threshold γ	0.5, 0.6, 0.7	0.5, 0.6, 0.7	0.5, 0.6, 0.7
Contrastive weight λ_1	$\{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}\}$	$\{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}\}$	$\{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}\}$
L2 weight λ_2	$\{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}\}$	$\{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}\}$	$\{10^{-5}, 10^{-4}, 10^{-3}, 10^{-2}\}$
Temperature τ	$\{0.3, 0.5, 1, 5, 10\}$	$\{0.3, 0.5, 1, 5, 10\}$	$\{0.3, 0.5, 1, 5, 10\}$

4.5. Performance Comparison

In this subsection, we evaluate the overall performance of the proposed AIHR-PCRM on three publicly real-world datasets and validate the effectiveness of our AIHR-PCRM model by comparing it with various baseline models. The comparison results are reported in Table 4, from which we have the following observations. All numbers are mean ± standard deviation over five seeds, and the rightmost column lists the relative im-

provement of AIHR-PCRM over HHCoR (the strongest baseline). All gains are statistically significant at $p < 0.01$.

(1) AIHR-PCRM has significant improvement over all the baselines on three datasets for all metrics. Compared with HHCoR, it improves by 11.71%, 10.75%, 9.75%, and 10.82% on XuetangX for R@10, N@10, R@20 and N@20, respectively. On MOOCCube, it improves by 11.57%, 10.84%, 9.45%, and 14.67% for R@10, N@10, R@20, and N@20, respectively. And on CNPC, it improves by 10.50%, 12.35%, 8.16% and 11.37% for R@10, N@10, R@20, and N@20, respectively. This demonstrates the effectiveness of the proposed model.

(2) Some sequential recommendation methods (i.e., MECF and ISRA) cannot perform better than AIHR-PCRM, which is specialized for MOOC recommendation. This is because MOOC recommendation is not just a sequence modeling problem, and learners' and courses' attributes and the interactions between learners and courses all play important roles in determining course selections. We also notice that our proposed method performs better than the static recommendation approaches (i.e., HHCoR and XSimGCL). These results demonstrate the importance of modeling learners' sequential behaviors rather than just considering their static interactions.

(3) GNN-based sequential methods (i.e., MFHCR, HGNN, and AIHR-PCRM) have a significant improvement over the RNN-based method (i.e., MECF). This is because RNN-based methods can only capture courses' short-term sequential relationships and lack the ability of modeling courses' long-term dependencies. Compared with them, GNN-based methods connect all the courses in a hypergraph, and represent each node by the representation of their adjacent neighbors, enabling them to learn to a course embedding globally. AIHR-PCRM outperforms MFHCR and HGNN, which again demonstrates the superiority of our CHR solution.

Beyond the numerical gains, we analyze why each design choice contributes theoretically.

(a) Why AIHR-PCRM substantially outperforms RNN-based methods (e.g., MECF): RNN-based methods are restricted to short-range sequential dependencies because of gradient decay over long sequences. AIHR-PCRM bypasses this locality bottleneck in two ways: CHR-derived hyperedges encode global course co-occurrence as one-hop neighbors in the hypergraph (constant-distance access regardless of sequence length), and BSM provides bidirectional attention that captures dependencies of arbitrary distance.

(b) Why AIHR-PCRM outperforms standard hypergraph methods (e.g., MFHCR, HHCoR): Standard hypergraph methods construct hyperedges directly from raw interactions, producing high redundancy and noise. CHR reconstructs the hypergraph by removing invalid learners and filtering redundant hyperedges, raising the signal-to-noise ratio of the hypergraph structure. The downstream HMP then propagates cleaner global collaborative signals, which explains the consistent gains across all three datasets.

(c) Why the improvement is largest on CNPC (e.g., 14.67% on N@20): CNPC has the densest average learner interaction (10.2) among the three datasets, which makes high-order reachable course sets more informative and CHR more effective. This explains the dataset-dependent magnitude of improvement.

Table 4. Performance comparison on XuetangX, MOOCCube, and CNPC (chronological leave-one-out split). All results are mean ± std over 5 seeds, and * denotes $p < 0.01$ vs. HHCoR.

Dataset	Metric	HGCR	MECF	Light-GCN	ISRA	HGNN
XuetangX	R@10	0.0654 ± 0.0025	0.0911 ± 0.0021	0.1403 ± 0.0027	0.4382 ± 0.0029	0.4964 ± 0.0028
	N@10	0.0537 ± 0.0019	0.0798 ± 0.0022	0.1024 ± 0.0025	0.2673 ± 0.0024	0.2935 ± 0.0027
	R@20	0.1632 ± 0.0025	0.2146 ± 0.0026	0.3518 ± 0.0031	0.5481 ± 0.0027	0.6674 ± 0.0033
	N@20	0.0614 ± 0.0023	0.0948 ± 0.0026	0.1256 ± 0.0019	0.2932 ± 0.0025	0.3290 ± 0.0029
MOOCCube	R@10	0.1001 ± 0.0035	0.1312 ± 0.0037	0.1874 ± 0.0042	0.3312 ± 0.0038	0.3528 ± 0.0045
	N@10	0.0731 ± 0.0027	0.0939 ± 0.0036	0.1216 ± 0.0029	0.2219 ± 0.0041	0.2396 ± 0.0034
	R@20	0.1662 ± 0.0037	0.2126 ± 0.0039	0.2851 ± 0.0040	0.4237 ± 0.0032	0.4510 ± 0.0045
	N@20	0.1129 ± 0.0035	0.1523 ± 0.0027	0.1831 ± 0.0038	0.2719 ± 0.0029	0.2654 ± 0.0040
CNPC	R@10	0.1136 ± 0.0051	0.1811 ± 0.0045	0.2415 ± 0.0042	0.2954 ± 0.0038	0.3916 ± 0.0049
	N@10	0.0615 ± 0.0047	0.1043 ± 0.0045	0.1695 ± 0.0038	0.1815 ± 0.0041	0.2263 ± 0.0037
	R@20	0.1925 ± 0.0044	0.2637 ± 0.0037	0.3233 ± 0.0048	0.3876 ± 0.0037	0.4743 ± 0.0040
	N@20	0.1396 ± 0.0032	0.1686 ± 0.0042	0.2216 ± 0.0045	0.2613 ± 0.0041	0.3114 ± 0.0056
Dataset	Metric	MFHCR	XSimGCL	HHCoR	AIHR-PCRM	Improvment
XuetangX	R@10	0.5134 ± 0.0026	0.6123 ± 0.0024	0.6216 ± 0.0031	0.6944 ± 0.0034	+11.71% * ($p < 0.001$)
	N@10	0.2997 ± 0.0019	0.3938 ± 0.0025	0.4129 ± 0.0024	0.4573 ± 0.0027	+10.75% * ($p < 0.001$)
	R@20	0.6921 ± 0.0026	0.7841 ± 0.0024	0.8016 ± 0.0029	0.8798 ± 0.0033	+9.75% * ($p < 0.001$)
	N@20	0.3376 ± 0.0021	0.4508 ± 0.0027	0.4721 ± 0.0026	0.5232 ± 0.0028	+10.82% * ($p < 0.001$)
MOOCCube	R@10	0.3715 ± 0.0036	0.4175 ± 0.0040	0.4872 ± 0.0042	0.5436 ± 0.0039	+11.57% * ($p < 0.001$)
	N@10	0.2431 ± 0.0031	0.2911 ± 0.0028	0.3439 ± 0.0033	0.3812 ± 0.0030	+10.84% * ($p < 0.001$)
	R@20	0.4723 ± 0.0028	0.5719 ± 0.0034	0.6519 ± 0.0036	0.7135 ± 0.0038	+9.45% * ($p < 0.001$)
	N@20	0.2893 ± 0.0032	0.3249 ± 0.0038	0.3675 ± 0.0035	0.4214 ± 0.0034	+14.67% * ($p < 0.001$)
CNPC	R@10	0.4242 ± 0.0032	0.5378 ± 0.0035	0.5537 ± 0.0048	0.6119 ± 0.0046	+10.50% * ($p < 0.01$)
	N@10	0.2871 ± 0.0047	0.3357 ± 0.0051	0.4418 ± 0.0042	0.4964 ± 0.0040	+12.35% * ($p < 0.01$)
	R@20	0.5492 ± 0.0051	0.6188 ± 0.0053	0.7512 ± 0.0040	0.8125 ± 0.0042	+8.16% * ($p < 0.01$)
	N@20	0.3818 ± 0.0038	0.4412 ± 0.0043	0.5036 ± 0.0038	0.5612 ± 0.0041	+11.37% * ($p < 0.01$)

4.6. Ablation Study

To evaluate the impact of the modules, we conduct ablation studies as shown in Table 5. The variants are as follows: (1) AIHR-PCRM: The proposed model. (2) AIHR-PCRM-w/o JEM: JEM removed. (3) AIHR-PCRM-w/o CHR: CHR removed. (4) AIHR-PCRM-w/o BSM: BSM disabled. (5) AIHR-PCRM-w/o JEM&CHR: Both JEM and CHR removed simultaneously.

(1) Validity of JEM: AIHR-PCRM outperforms AIHR-PCRM-w/o JEM by 3.81%/6.86% on XuetangX R@20/N@20, 4.59%/9.09% on MOOCCube, and 8.87%/14.04% on CNPC. This indicates that attribute encoding helps the model perceive specific features of learners and courses.

(2) Validity of CHR: AIHR-PCRM-w/o CHR drops by 9.08%/20.58%, 12.91%/27.12%, and 20.92%/36.11% on the three datasets for R@20/N@20, demonstrating that HGM with CHR captures high-quality global collaborative relationships and alleviates sparsity.

(3) Validity of BSM: AIHR-PCRM-w/o BSM drops by 6.98%/9.80% on XuetangX, 7.73%/16.51% on MOOCCube, and 13.83%/28.98% on CNPC, confirming that bidirectional attention captures long-range dependencies.

(4) Joint contribution of JEM and CHR: On XuetangX R@20, removing JEM alone drops performance by 3.67% and removing CHR alone drops by 8.92%, summing to 12.59%. The newly added variant AIHR-PCRM-w/o JEM&CHR drops by 18.32%, which exceeds the sum of the two individual drops. The same super-additive pattern holds on MOOCCube (combined drop 22.59% vs. sum-of-singles 16.83%) and CNPC (combined drop 27.68% vs. sum-of-singles 25.85%). This provides empirical support that JEM and CHR/HGM are jointly beneficial rather than merely additive: the two modules co-adapt through the shared gradient pathway derived, so disabling one already reduces the effectiveness of the other.

Table 5. Ablation study (mean ± std over 5 seeds).

Variants	XuetangX R@20	XuetangX N@20	MOOCCube R@20	MOOCCube N@20	CNPC R@20	CNPC N@20
AIHR-PCRM	0.8798 ± 0.0033	0.5232 ± 0.0028	0.7135 ± 0.0038	0.4214 ± 0.0034	0.8125 ± 0.0042	0.5612 ± 0.0041
AIHR-PCRM-w/o JEM	0.8475 ± 0.0036	0.4896 ± 0.0030	0.6822 ± 0.0041	0.3863 ± 0.0037	0.7463 ± 0.0044	0.4921 ± 0.0044
AIHR-PCRM-w/o CHR	0.8013 ± 0.0040	0.4339 ± 0.0034	0.6319 ± 0.0044	0.3315 ± 0.0040	0.6719 ± 0.0049	0.4123 ± 0.0048
AIHR-PCRM-w/o BSM	0.8224 ± 0.0038	0.4765 ± 0.0032	0.6623 ± 0.0042	0.3617 ± 0.0038	0.7138 ± 0.0046	0.4351 ± 0.0046
AIHR-PCRM-w/o JEM&CHR	0.7186 ± 0.0049	0.4012 ± 0.0042	0.5523 ± 0.0051	0.2941 ± 0.0044	0.5876 ± 0.0058	0.3528 ± 0.0053

4.7. Data Sparsity Study

To further test the impact of data sparsity on the performance of the proposed AIHR-PCRM model, a series of experiments with different data sparsities are conducted. Learners are categorized into groups based on the number of interactions they have with the course. For example, the first group in the learner-side experiments contains learners interacting with 0–5 courses. The summarized results are compared with three representative baseline models, ISRA, MFHCR, and HHCoR in Figure 3. Obviously, the hypergraph global collaborative learning module (HGM) can capture implicit global collaboration relationships to solve the sparsity issue of the learner behavior sequence. It can also be observed that ISRA, MFHCR, and HHCoR perform relatively unstably at different data sparsities. This suggests that ISRA, MFHCR, and HHCoR may not be able to capture high-quality learner preference representations from sparse interaction data.

4.8. Impact of Model Hyperparameters

In this section, we study the impact of several key hyperparameters (i.e., hyperedge filtering ratio parameter γ and temperature hyperparameter τ) in our AIHR-PCRM and report the evaluation results in Figures 4 and 5.

(1) Impact of the hyperparameter τ on the proposed model performance: In the contrastive learning framework, the hyperparameter τ controls the strength of the contrastive loss to recognize hard negative samples. From the evaluation results in Figure 4, it can be observed that the best recommended performance is obtained by using $\tau = 1.0$. A larger value of $\tau(\tau > 1.0)$ makes the gradient of learning hard negative samples become smaller, which leads to a degradation of the recommendation performance.

(2) Impact of the hyperparameter γ on the proposed model performance: As shown in Figure 5, when the value of γ is at 0.6, the performance of the model on three datasets can reach a better level. This is because the AIHR-PCRM model was initially modeled to capture diverse and high-quality implicit collaborative signals between courses.

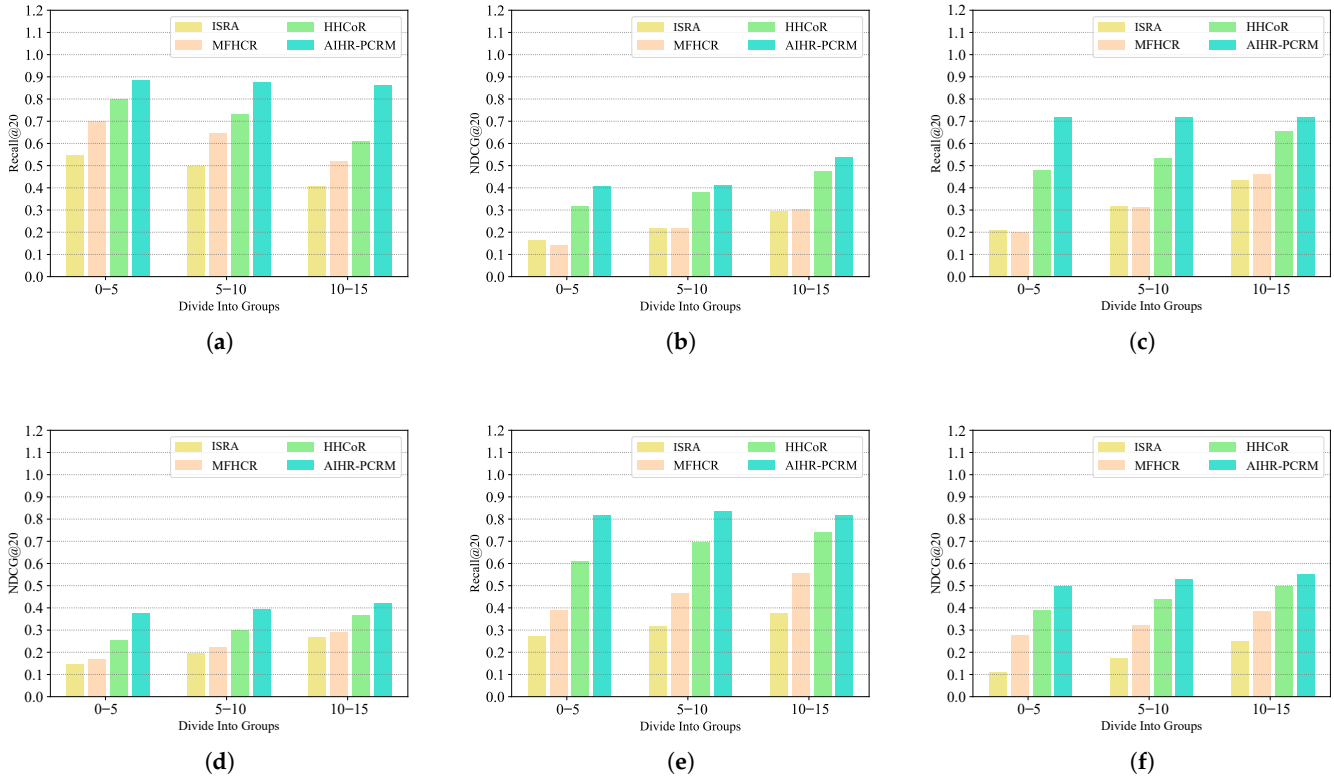

Figure 3. Performance with different data sparsity degrees on XuetangX (**a**,**b**), MOOCCube (**c**,**d**), and CNPC (**e**,**f**) datasets.

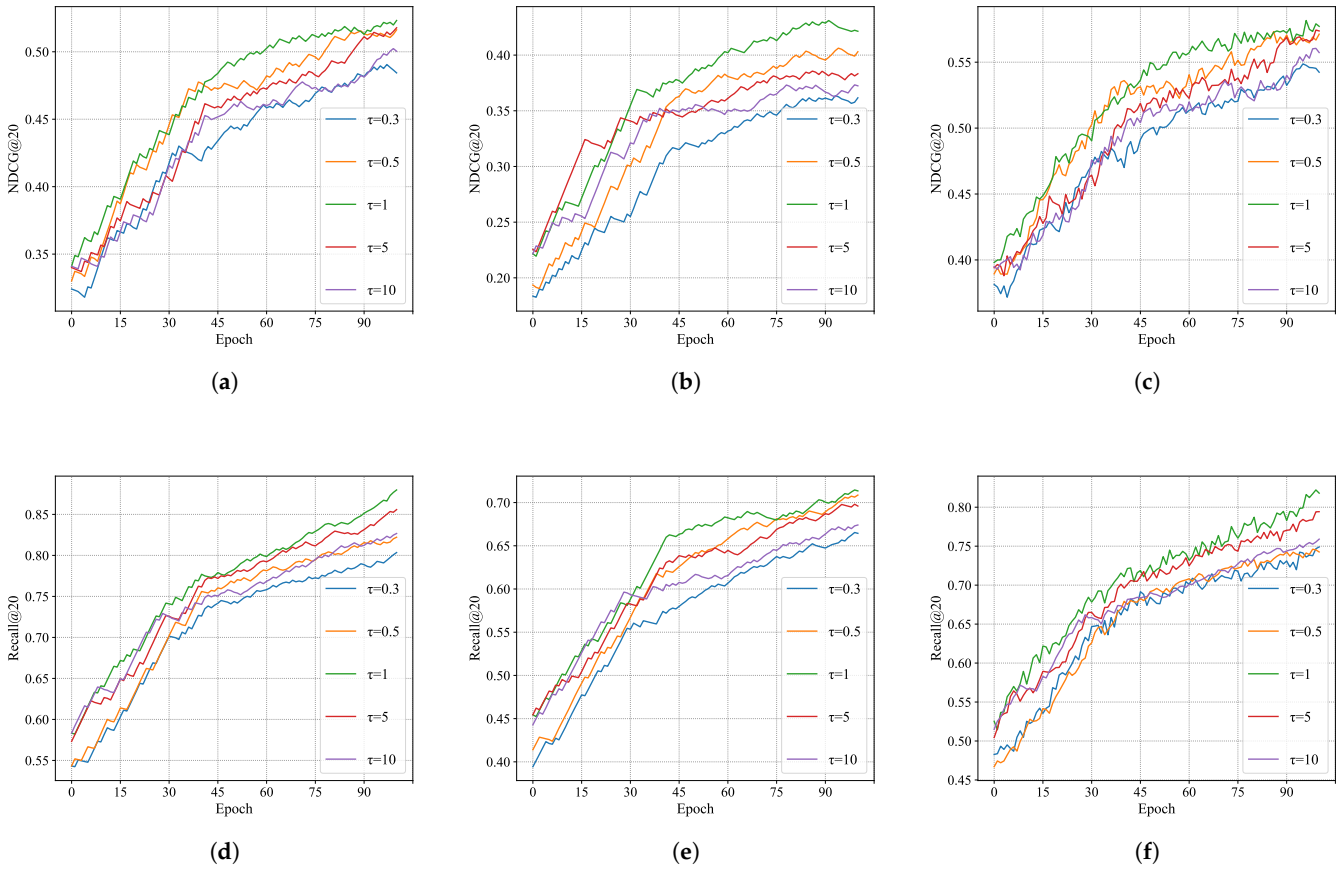

Figure 4. The influence of hyperparameter τ on XuetangX (**a**,**d**), MOOCCube (**b**,**e**), and CNPC (**c**,**f**) datasets.

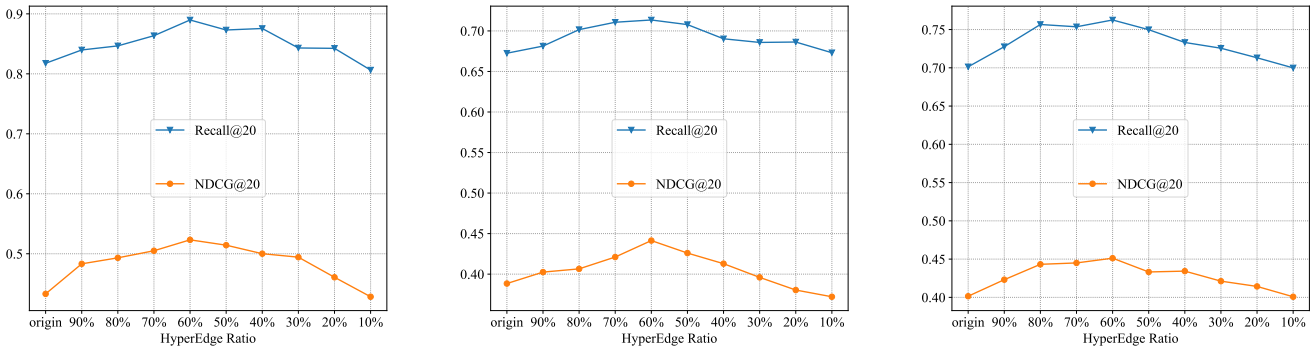

Figure 5. The influence of hyperedge ratio on XuetangX (**a**), MOOCCube (**b**), and CNPC (**c**) datasets.

5. Conclusions

In this work, we propose the AIHR-PCRM model to alleviate the issue of data sparsity for learners' behavior sequences, as well as the attributes of the course that are not comprehensively studied in the existing course recommendation methods. The model utilizes the course hypergraph reconstruction (CHR) technique to construct the course hypergraph structure and captures the implicit global dependencies between courses through a hypergraph message-passing paradigm to alleviate the data sparsity issue. Learner and course attribute representations are also jointly encoded with course hyper-embeddings to capture richer learner behavioral features. Experimental results based on the XuetangX, MOOCCube, and CNPC datasets show that the personalized course recommendation performance of the AIHR-PCRM model outperforms that of the existing frontier baseline model. We identify two promising directions for future work. (i) Attribute-driven feedback between modules: The current AIHR-PCRM fuses attribute, sequence, and hyperedge information additively. A natural extension is to allow attribute embeddings to feed back and modulate the weights of specific hyperedges, e.g., through an attention gate. Our preliminary experiments (Appendix A) show that this direction is technically feasible but introduces training-stability challenges that warrant further investigation. (ii) Multi-learner behavioral interactions: The impact of cooperative/competitive interactions among learners on course recommendation remains an open research direction.

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

Funding: This work was supported in part by the National Natural Science Foundation of China under grant 62373069, Natural Science Foundation of Chongqing under grant CSTB2023NSCQ-LZX0094, and Science and Technology Research Program of Chongqing Municipal Education Commission under Grants 232130, and YJG232045.

Data Availability Statement: Data availability status: Data available in a publicly accessible repository. Recommended Data Availability Statement: The original data presented in the study are openly available in XuetangX [36], MOOCCube [37], CNPC in http://thedata.harvard.edu/ (accessed on 23 October 2025).

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

Appendix A. Preliminary Feedback-Mechanism Experiments

To incorporate inter-module feedback, we implemented an attention-based gate where attribute embeddings modulate the hypergraph embeddings before joint encoding: $\tilde{\delta}_{i,f}^{(c)} \leftarrow \sigma\left(W \cdot \left[M_{c_i^{(s)}}; M_s\right]\right) \odot \delta_{i,f}^{(c)}$. On MOOCCube R@20, this variant achieved a marginal improvement of +0.6% (0.7178 vs. 0.7135) but reduced training stability (validation NDCG std doubled from 0.004 to 0.009) and increased convergence time by 35%. We therefore retain the additive fusion form as the main model but report this preliminary result for completeness and as motivation for future work.

Appendix B. Robustness Under Global Temporal Split

As a complementary robustness check beyond the per-user chronological split used in the main experiments, we re-evaluate all methods under a strict global temporal split: all interactions before timestamp t_{train} serve as training data, those between t_{train} and t_{val} serve as validation data, and those after t_{val} serve as test data (using the 70%/20%/10% temporal quantiles per dataset). Although absolute scores drop slightly for all methods (due to harder generalization to entirely future time periods), the relative ranking is preserved and AIHR-PCRM continues to outperform all baselines across all metrics.

References

1. Liu, Y.; Dong, Y.; Yin, C. A Personalized Course Recommendation Model Integrating Multi-granularity Sessions and Multi-type Interests. *Educ. Inf. Technol.* **2024**, *29*, 5879–5901. [CrossRef]
2. Shaheen, M.; Ghafoor, R.; Sugathan, S.K.; Isawasan, P.; Asmawi, M.A.H.A. Unveiling the Factors for MOOC Adoption: An Educational Data Mining Perspective. *Information* **2026**, *17*, 175. [CrossRef]
3. Zheng, Y.; Wang, D.; Zhang, J. A unified framework for personalized learning pathway recommendation in e-learning contexts. *Educ. Inf. Technol.* **2025**, *30*, 7911–7948. [CrossRef]
4. Majjate, H.; Bellarhmouch, Y.; Jeghal, A. Assessing the impact of ethical aspects of recommendation systems on student trust and engagement in E-learning platforms: A multifaceted investigation. *Educ. Inf. Technol.* **2025**, *30*, 3953–3977. [CrossRef]
5. Delianidi, M.; Diamantaras, K.; Kokkonis, G.; Sidiropoulos, A.; Evangelidis, G.; Karapiperis, D. DK-PRACTICE: An Intelligent Platform for Knowledge Tracing and Educational Content Recommendation: A Case Study in Higher Education. *Information* **2026**, *17*, 202. [CrossRef]
6. Yao, W.; Hu, X. What learns next: Learning intents guided dual contrastive learning model for online course recommendation. *Neurocomputing* **2025**, *637*, 130051. [CrossRef]
7. Zhang, G.; Gao, X.; Ye, H.; Zhu, J.; Lin, W.; Wu, Z.; Zhou, H.; Ye, Z.; Ge, Y.; Baghban, A. Optimizing learning paths: Course recommendations based on graph convolutional networks and learning styles. *Appl. Soft Comput.* **2025**, *175*, 113083. [CrossRef]
8. Xu, J.; Chen, Z.; Ma, Z.; Liu, J.; Ngai, E.C.H. Improving Consumer Experience With Pre-Purify Temporal-Decay Memory-Based Collaborative Filtering Recommendation for Graduate School Application. *IEEE Trans. Consum. Electron.* **2025**, *71*, 5783–5791. [CrossRef]
9. Yu, X.; Mao, Q.; Wang, X.; Yin, Q.; Che, X.; Zheng, X. CR-LCRP: Course recommendation based on Learner–Course Relation Prediction with data augmentation in a heterogeneous view. *Expert Syst. Appl.* **2024**, *249*, 123777. [CrossRef]
10. Balaji, V.; Anupam, D.; Vishnupriya, G.; Safak, K. Enhancement of single candidate optimizer for weighted feature fusion and dilation-based cascaded RNN in learning-based recommendation system. *Knowl.-Based Syst.* **2025**, *329*, 114319.
11. Sun, J.; Mei, S.; Yuan, K.; Jiang, Y.; Cao, J. Prerequisite-enhanced category-aware graph neural networks for course recommendation. *ACM Trans. Knowl. Discov. Data* **2024**, *18*, 1–21. [CrossRef]
12. Zhao, Y.; Zheng, Y. MOOC Recommendation Using Heterogeneous Graph Neural Network and Attention Mechanism. In Proceedings of the 2023 6th International Conference on Artificial Intelligence and Pattern Recognition, Haikou, China, 18–20 August 2023; pp. 1376–1381.
13. Wang, J.; Ding, K.; Hong, L.; Liu, H.; Caverlee, J. Next-item recommendation with sequential hypergraphs. In Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual, 25–30 July 2020; pp. 1101–1110.

14. Li, X.; Zhang, Y.; Huang, Y.; Li, K.; Zhang, Y.; Wang, X. Multi-aspect Knowledge-enhanced Hypergraph Attention Network for Conversational Recommendation Systems. *Knowl.-Based Syst.* **2024**, *299*, 112119. [CrossRef]
15. Zhao, Y.; Jiang, F.; Pang, Y.; Deng, Y.; Han, Y.; Wang, J. EduLGCL: Local-global contrastive learning model for education recommendation. *Knowl.-Based Syst.* **2024**, *286*, 111357. [CrossRef]
16. Mahmood, S.; Hasan, R.; Ahmad, S. HSE-GNN-CP: Spatiotemporal Teleconnection Modeling and Conformalized Uncertainty Quantification for Global Crop Yield Forecasting. *Information* **2026**, *17*, 141. [CrossRef]
17. He, X.; Deng, K.; Wang, X.; Li, Y.; Zhang, Y.; Wang, M. Lightgcn: Simplifying and powering graph convolution network for recommendation. In Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual, 25–30 July 2020; pp. 639–648.
18. Liu, X.; Meng, S.; Li, Q.; Liu, Q.; He, Q.; Ramesh, D.; Qi, L. Fdgnn: Feature-aware disentangled graph neural network for recommendation. *IEEE Trans. Comput. Soc. Syst.* **2023**, *11*, 1372–1383. [CrossRef]
19. Seo, C.; Jeong, K.J.; Lim, S.; Shin, W.Y. SiReN: Sign-aware recommendation using graph neural networks. *IEEE Trans. Neural Netw. Learn. Syst.* **2022**, *35*, 4729–4743. [CrossRef]
20. Yu, J.; Xia, X.; Chen, T.; Cui, L.; Hung, N.Q.V.; Yin, H. XSimGCL: Towards extremely simple graph contrastive learning for recommendation. *IEEE Trans. Knowl. Data Eng.* **2024**, *36*, 913–926. [CrossRef]
21. Li, M.; Li, Z.; Huang, C.; Jiang, Y.; Wu, X. EduGraph: Learning Path-based Hypergraph Neural Networks for MOOC Course Recommendation. *IEEE Trans. Big Data* **2024**, *accepted*.
22. Jiang, L.; Xiao, Y.; Zhao, X.; Xu, Y.; Hu, S.; Wang, P.; Yin, M. Hierarchical Reinforcement Learning on Multi-Channel Hypergraph Neural Network for Course Recommendation. In Proceedings of the Thirty-Third International Joint Conference on Artificial Intelligence, IJCAI-24, Jeju, Republic of Korea, 3–9 August 2024; pp. 2099–2107.
23. Wang, X.; Ma, W.; Guo, L.; Jiang, H.; Liu, F.; Xu, C. HGNN: Hyperedge-based graph neural network for MOOC course recommendation. *Inf. Process. Manag.* **2022**, *59*, 102938. [CrossRef]
24. Sun, A.; Yang, K.; Ren, D.; Wang, X. Online Course Recommendation with Hypergraph-based Multi-Channel Feature Fusion. In Proceedings of the 2023 6th International Conference on Artificial Intelligence and Pattern Recognition, Haikou, China, 18–20 August 2023; pp. 1491–1497.
25. Zhang, H.; Shen, X.; Yi, B.; Wang, W.; Feng, Y. KGAN: Knowledge grouping aggregation network for course recommendation in MOOCs. *Expert Syst. Appl.* **2023**, *211*, 118344. [CrossRef]
26. Huang, S.; Dong, Y.; Wang, Z.; Zhou, N.; Ping, Y. MRTI-CR: A model based on multi-relationship and time-aware interest for personalized course recommendation. *Eng. Appl. Artif. Intell.* **2025**, *159*, 111560. [CrossRef]
27. Yang, Q.; Li, Z.; Wu, Z.; Huang, Y.; Zhang, J. Multi-behavior multivariate contrastive learning for MOOC video recommendation. *Neurocomputing* **2025**, *655*, 131416. [CrossRef]
28. Wang, Y.; Ma, D.; Ma, J.; Jin, Q. HGCR: A Heterogeneous Graph-Enhanced Interactive Course Recommendation Scheme for Online Learning. *IEEE Trans. Learn. Technol.* **2024**, *17*, 364–374. [CrossRef]
29. Li, S.; Zhao, Y.; Guo, L.; Ren, M.; Jin, L.; Zhang, L.; Li, K. Quantification and prediction of engagement: Applied to personalized course recommendation to reduce dropout in MOOCs. *Inf. Process. Manag.* **2024**, *61*, 103536. [CrossRef]
30. Zhai, X.; Wang, Y.; Liang, L.; Wang, K.; Pei, F.; Fu, E.Y. Personalized e-learning resource recommendation using multimodal-enhanced collaborative filtering. *Knowl.-Based Syst.* **2025**, *319*, 113605. [CrossRef]
31. Wang, B.; Li, P. Personalized recommendation based on improved speech recognition algorithm in music e-learning course simulation. *Entertain. Comput.* **2025**, *52*, 100721. [CrossRef]
32. Asri, B.; Qassimi, S.; Rakrak, S. Adaptive Personalized Recommendation Systems: A systematic Review. *Inf. Syst.* **2026**, *135*, 102594. [CrossRef]
33. Xia, L.; Huang, C.; Xu, Y.; Zhao, J.; Yin, D.; Huang, J. Hypergraph Contrastive Collaborative Filtering. In Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR'22), Madrid, Spain, 11–15 July 2022; ACM: New York, NY, USA, 2022; pp. 70–79. [CrossRef]
34. Sun, A.; Lu, J. Contrastive Enhanced Filter Model Based on Bidirectional Transformer for Sequential Recommendation. In Proceedings of the 2024 Asia-Pacific Conference on Image Processing, Electronics and Computers (IPEC), Dalian, China, 12–14 April 2024; pp. 658–663.
35. Singh, N.K.; Tomar, D.S.; Shabaz, M.; Keshta, I.; Soni, M.; Sahu, D.R.; Bhende, M.; Nandanwar, A.K.; Vishwakarma, G. Self-Attention Mechanism Based Federated Learning Model for Cross Context Recommendation System. *IEEE Trans. Consum. Electron.* 2024, *accepted*.

36. Gong, J.; Wan, Y.; Liu, Y.; Li, X.; Zhao, Y.; Wang, C.; Li, Q.; Feng, W.; Tang, J. Reinforced MOOCs Concept Recommendation in Heterogeneous Information Networks. *arXiv* **2022**, arXiv:2203.11011. [CrossRef]
37. 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, Online, 5–10 July 2020; pp. 3135–3142.