

Article

Enhancing Knowledge-Concept Recommendations with Heterogeneous Graph-Contrastive Learning

Litong Wei ^{1,2} , Yun Li ^{2,*}, Weiwei Wang ² and Yi Zhu ²

¹ Jiangsu Provincial Key Constructive Laboratory for Big Data of Psychology and Cognitive Science, Yancheng Teachers University, Yancheng 224002, China; weil@yctu.edu.cn

² School of Information Engineering, Yangzhou University, Yangzhou 225012, China; yangzhouwangweiwei@163.com (W.W.); zhuyi@yzu.edu.cn (Y.Z.)

* Correspondence: liyun@yzu.edu.cn; Tel.: +86-138-5278-5198

Abstract: With the implementation of conceptual labeling on online learning resources, knowledge-concept recommendations have been introduced to pinpoint concepts that learners may wish to delve into more deeply. As the core subject of learning, learners' preferences in knowledge concepts should be given greater attention. Research indicates that learners' preferences for knowledge concepts are influenced by the characteristics of their group structure. There is a high degree of homogeneity within a group, and notable distinctions exist between the internal and external configurations of a group. To strengthen the group-structure characteristics of learners' behaviors, a multi-task strategy for knowledge-concept recommendations is proposed; this strategy is called Knowledge-Concept Recommendations with Heterogeneous Graph-Contrastive Learning. Specifically, due to the difficulty of accessing authentic social networks, learners and their structural neighbors are considered positive contrastive pairs to construct self-supervision signals on the predefined meta-path from heterogeneous information networks as auxiliary tasks, which capture the higher-order neighbors of learners by presenting different perspectives. Then, the Information Noise-Contrastive Estimation loss is regarded as the main training objective to increase the differentiation of learners from different professional backgrounds. Extensive experiments are constructed on MOOCCube, and we find that our proposed method outperforms the other state-of-the-art concept-recommendation methods, achieving 6.66% with $HR@5$, 8.85% with $NDCG@5$, and 8.68% with MRR .



Citation: Wei, L.; Li, Y.; Wang, W.; Zhu, Y. Enhancing Knowledge-Concept Recommendations with Heterogeneous Graph-Contrastive Learning.

Mathematics **2024**, *12*, 2324. <https://doi.org/10.3390/math12152324>

Academic Editor: Andrea Scozzari

Received: 17 June 2024

Revised: 21 July 2024

Accepted: 23 July 2024

Published: 25 July 2024



Copyright: © 2024 by the authors. Licensee MDPI, Basel, Switzerland. This article is an open access article distributed under the terms and conditions of the Creative Commons Attribution (CC BY) license (<https://creativecommons.org/licenses/by/4.0/>).

Keywords: knowledge-concept recommendation; group-structure feature of learner preferences; graph-contrastive learning

MSC: 68Txx

1. Introduction

Online learning platforms, including MOOCs (massive open online courses), have revolutionized the way people learn by offering a flexible and accessible mode of education. They have played a crucial role during the pandemic by providing an alternative mode of education that enables learners to continue their education without being physically present in a classroom [1,2]. XuetangX [3,4] is one of the largest MOOC platforms in China, and it has gained popularity among learners due to its effective and fine-grained course concepts that help students understand course material in a faster and more efficient way. Additionally, a large-scale data repository, MOOCCube [5], has been made available by XuetangX to facilitate natural language processing (NLP) and artificial intelligence (AI) research for educational applications.

Knowledge-concept recommendations are designed to predict concepts that learners might be interested in according to their historical learning behaviors. Previous research studies have demonstrated that [4,6] have indicated that learners from varying professional backgrounds tend to focus on distinct knowledge concepts even when taking the same

courses with different teachers. Moreover, it has been found that clicking on relevant knowledge concepts can enhance learning progress and speed. By taking a more microscopic view of a learner's needs, knowledge-concept recommendations [7,8] can provide more fine-grained services and experiences compared to course recommendations, ultimately enhancing the overall online learning experience.

Collaborative filtering (CF) has been widely applied to recommendation tasks; it involves mapping users and items to a latent representation space and preserving the structural feature information of user-item interactions. It is also an effective method for solving the problem of concept recommendations. However, as the number of learners and concepts increases, the sparsity of interactions becomes a challenging issue. To alleviate the issue of data sparsity, most knowledge-concept recommendation methods have constructed a heterogeneous information network (HIN) on MOOC platforms. The approach helps enrich the representation of interaction entities and improve recommendation performance [4,9,10].

In real-world educational settings, learners' behaviors are influenced by the characteristics of group structures. As illustrated in Figure 1, the MOOCCube dataset includes 50 students selected from three different academic backgrounds: medicine, chemistry, and the dialectics of nature. Students with a background in medicine numbered 0–9, those in chemistry numbered 10–46, and those in the dialectics of nature numbered 47–49. The heatmap shows the probabilities representing the intersection and union ratios of the students' clicks on knowledge concepts. It is evident from the heatmap that students' preferences for knowledge concepts are highly similar within the same group structure, and there are noticeable differences between different group structures. Knowledge-concept recommendation methods based on heterogeneous graphs focus on similarities within group structures, and they extract features from learners in various dimensional neighborhoods to enrich learner representations.

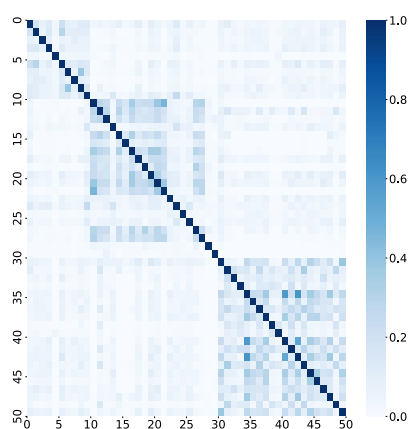


Figure 1. MOOCCube dataset behavior heatmap showing probability as the proportion of learners' clicks on the intersection and union of knowledge concepts.

However, two key limitations are hindering further development in concept recommendations: (1) Due to data sparsity, the similar relationships within group structures cannot be explicitly utilized to enrich graph information when using a graph convolution-based collaborative filtering method, resulting in the insufficient mining of learner characteristics. (2) Ignoring the differences between group structures undoubtedly affects the capture of learner preferences, leading to biased results in knowledge-concept recommendations.

To address the aforementioned challenges in knowledge concept-recommendation tasks, a novel multi-task method called Knowledge Concept Recommendation with Heterogeneous Graph-Contrastive Learning (KCRHGCL) is proposed in this paper. The goal of KCRHGCL is to enhance learner representations by extracting features of similarities and differences in group structures, thereby improving the effectiveness of knowledge-concept recommendations. "Birds of a feather flock together". Learners' neighbors can enhance their feature representations. Intuitively, Structural neighbors, by considering the local

structure of learners in the graph, capture subtle differences and similarities in their features, thereby enriching the characteristic features of learners' preferences. Specifically, by using a structure-aware contrastive learning method on each predefined meta-path, the representations of learners are brought closer to the embedding of their structural neighbors, thereby enhancing the learners' representations. The structural neighbors of a learner are defined as the embeddings of the learner based on the output of the graph convolutional layer. These self-supervised tasks are treated as auxiliary tasks to improve the recommendation performance. In addition, negative sampling techniques are used to distinguish features of different group structures by selecting negative samples that are more likely to be confused with positive samples. By combining this with InfoNCE, our proposed method can effectively maximize the mutual information between similar learners and minimize it between dissimilar learners, thereby enhancing the overall recommendation performance.

In summary, this paper makes the following contributions:

- (1) A new method called the Heterogeneous Graph-Contrastive Learning for Knowledge concept recommendation is proposed. It incorporates structure-aware contrastive learning to mine higher-order similarities within the learner-group structure while aggregating learner-neighbor features from different dimensions, more fully capturing the preferences of learners.
- (2) Data analysis finds that, unlike commercial recommendations, learner behavior in knowledge-concept recommendations shows significant differences between group structures due to the influence of professional backgrounds. Negative sampling techniques better learn to distinguish features of different group structures by selecting negative samples that are more likely to be confused with the positive samples. InfoNCE loss enhances the differences between groups by maximizing the similarity of positive sample pairs and minimizing the similarity between positive and negative samples, thereby capturing the differences between different group structures.
- (3) Comprehensive experiments are conducted on a MOOCube dataset to validate the effectiveness of KCRHGCL. The results demonstrate that KCRHGCL outperforms other state-of-the-art methods.

2. Materials and Methods

2.1. Recommender Systems in MOOCs

MOOC platforms have seen a huge influx of learning resources, and the ever-increasing amount of material has left learners overwhelmed, unable to ensure the quality of online learning and leading to a loss of interest and persistently high dropout rates. Researchers have made numerous attempts, such as learner profiles and course recommendations. Currently, course recommendations [7,11,12] are one of the most widely applied tools. Collaborative filtering is one of the common techniques of course recommendation; it maps learners and courses to a latent space and captures learners' preferences. Some studies have shown that enhancing the characteristics of learners and courses can improve the effectiveness of course recommendations. For example, Jing et al. [13] introduced a course recommendation method that uses a collaborative filtering strategy incorporating user interests, demographic profiles, and course prerequisite relations. Zhang et al. [7] employed a hierarchical reinforcement learning algorithm to revise user profiles, and they combined it with the basic collaborative filtering model to improve the course recommendation task. Jung et al. [14] included a level-embedding module for students and courses in their course recommendation system, excluding the incorporation of a knowledge graph containing internal information about MOOCs and an external knowledge base. Knowledge concepts are smaller instructional units than courses, and they can be integrated according to different domains or topics. Using them as the target of recommendations allows for the tracking of learners' preferences from a more detailed perspective. As learning resources on online platforms are tagged with knowledge-concept labels, knowledge-concept recommendations have emerged. For example, Gong et al. [4] leverage an attention-based graph convolutional network to obtain representations of meta-paths that can capture

the multivariate preferences of learners. Building upon this work, Piao et al. [9] made three improvements: using implicit feedback instead of ratings, exploring more advanced attention mechanisms, and estimating the preference score of a concept in the prediction layer. Gong et al. [10] took token-modeling of the dynamic interactions among learners and concepts into consideration by treating concept recommendations as the reinforcement learning problem. However, learners' preferences are influenced by their professional backgrounds, which naturally form group characteristics. There is a similarity within these group structures and significant differences between the groups.

2.2. Recommendations with Contrastive Learning

Self-supervised learning [15,16] extracts transferable knowledge directly from the data itself without the need for labeled data, providing a new learning paradigm for addressing the data sparsity issues of interaction-based collaborative filtering methods. Contrastive learning, due to its lightweight model and flexible design, has gradually become the leading method recommended for self-supervised learning. The idea of CL is to maximize the agreement between different views, and the agreement is usually measured via mutual information [17]. Recently, many CL paradigms have been developed for interaction-based recommendation systems to enhance recommendation performance. For example, Xie et al. [18] applied a CL framework that tries three approaches to data augmentation to extract user representations based only on users' interactions. Yang et al. [19] proposed a novel approach that involves using cross-view self-discrimination supervision signals, along with a knowledge-aware contrastive objective, to improve recommendation robustness and address issues related to data noise and sparsity. Cai et al. [20] proposed the graph-contrastive learning paradigm LightGCN, which uses singular value decomposition for contrastive augmentation, allowing for the unconstrained refinement of the structure of global collaborative relationships. Lin et al. [21] introduced graph structure and semantic neighbors as contrastive pairs to enrich the representation of a current node. The aforementioned contrastive learning-based recommendation methods are based on homogeneous graphs. However, heterogeneous information networks can accurately distinguish between different semantics in the network by analyzing various types of nodes and the links between them, and they can mine the potential value and more meaningful knowledge from these rich objects and connections, making them widely used in recommendation systems. The application of contrastive learning to heterogeneous graphs has attracted the attention of researchers. Chen et al. [22] proposed a method called Heterogeneous Graph-Contrastive Learning for Recommendations, which incorporates heterogeneous relational semantics into user-item interaction modeling and enhances the knowledge transfer across different views using contrastive learning, and it is further enhanced with meta-networks for personalized and adaptive contrastive augmentation. Zheng et al. [23] proposed a method called Heterogeneous Information Crossing on Graphs, which captures both current interests and long-term preferences by constructing heterogeneous graphs from various user behaviors, and it incorporates contrastive learning to improve item representation and recommendation performance by leveraging items' co-occurrence across sessions. Inspired by these works, we proposed multi-task, meta-path, heterogeneous graph-contrastive learning to enhance learners' representations from multiple perspectives for knowledge-concept recommendations.

3. Preliminaries

Problem statement. Knowledge-concept recommendations aim to automatically recommend relevant knowledge concepts that learners might want to click on based on their historical behavior. Firstly, $\mathcal{L} = \{L_1, L_2, \dots, L_L\}$, $\mathcal{C} = \{C_1, C_2, \dots, C_C\}$, $\mathcal{V} = \{V_1, V_2, \dots, V_V\}$, $\mathcal{T} = \{T_1, T_2, \dots, T_T\}$, and $\mathcal{K} = \{K_1, K_2, \dots, K_K\}$ were defined to represent the set of learners, courses, videos, teachers, and knowledge concepts, respectively. **Input:** for the learners' \mathcal{U} and knowledge concepts \mathcal{K} , their corresponding interactive data \mathcal{R} , which can reveal relations from different meta-paths, $\mathcal{MP} = \{mp_1, mp_2, \dots, mp_M\}$, based on a heterogeneous information network, the details are described in the next section.

Output: a predict function, f , generates N knowledge-concept recommendation list, S (e.g., “osteoarthritis”, “quantum theory”, etc.), such that $f : l \rightarrow \{K_i \mid K_i \in \mathcal{K}, i < \mathcal{K}\}$.

Meta-paths based on the heterogeneous information network. On the MOOCs platform, interactive data, \mathcal{R} , can be organized as a heterogeneous information network (HIN), $\mathcal{G} = \{\mathcal{N}, \mathcal{E}\}$, where $\mathcal{N} = \{\mathcal{L} \cup \mathcal{C} \cup \mathcal{V} \cup \mathcal{T} \cup \mathcal{K}\}$ denotes the sets of objects, $\mathcal{E} = \{\{E_i\} \mid E_i \in \mathcal{O}\}$ denotes the set of links, $\mathcal{O} = \{\{Teach, Learn, Watch, Next, Include\}\}$, and $|\mathcal{N}| + |\mathcal{E}| > 2$. Meta-paths [24] are defined as paths in the form of a sequence of entity types, \mathcal{V} , connected by relationship types, \mathcal{E} , one of which is formally expressed as follows: $V_1 \xrightarrow{E_1} V_2 \xrightarrow{E_2} \dots \xrightarrow{E_l} V_{l+1}$.

4. Proposed Methods

First, the representation learning of the learners is introduced in Section 4.1. The final outputs contain the structural relations between learners on different meta-paths, aggregating through the layer propagation of GCN. Figure 2 illustrates the method flow of the overall framework. We first introduce the representation learning of the learners in Section 4.1, whose final outputs contain the structural relations between learners on different meta-paths aggregating through the layer propagation of GCN. Then, the representation learning of the concepts is introduced in Section 4.2, which uses the heterogeneous graph attention network to capture the representations from meta-paths. Finally, we describe a multi-task learning strategy in Section 4.3.

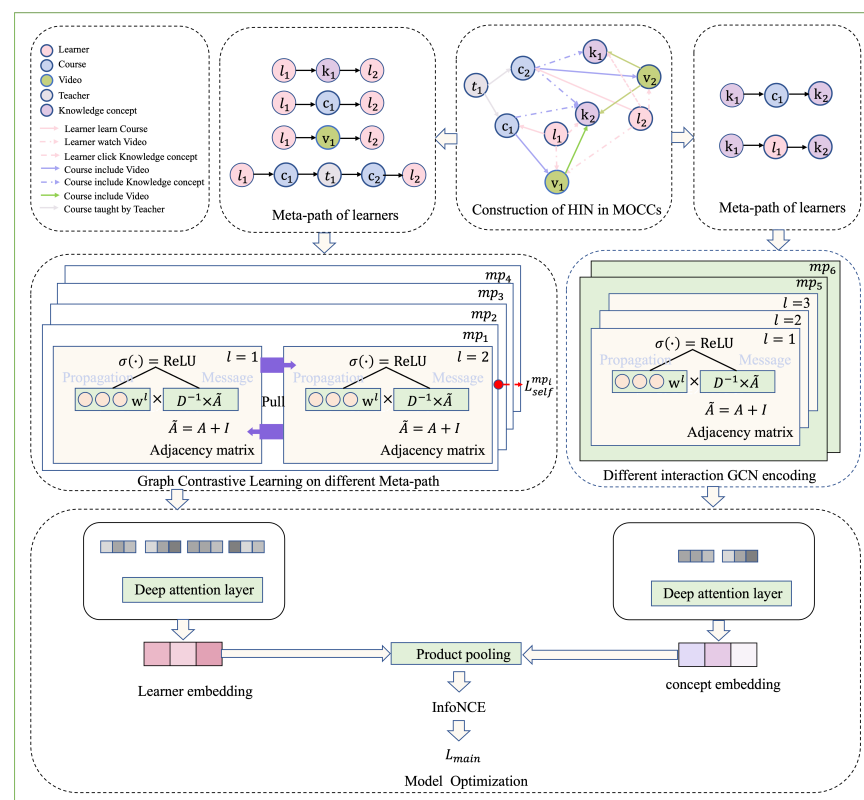


Figure 2. The method flow of the KCRHGCL framework, which consists of the following three main workflows: (i) the representation learning of the learners—in the constructed heterogeneous graph, appropriate meta-paths are selected, graph convolutional networks are used to extract features on these predefined meta-paths, and structural neighbors are introduced to enhance learner representations; (ii) the representation learning of the concepts—heterogeneous graph attention networks are used to capture the representations of concepts from different meta-paths; and (iii) the model optimization of the multi-task—negative sampling techniques and Information Noise Contrastive Estimation (InfoNCE) are utilized as the training objective loss to more effectively distinguish between differences in learners’ preferences across group structures.

4.1. The Representation Learning of the Learners

Meta-paths for learners selected from HIN in MOOCs. Figure 2 (The construction of an HIN in MOOCs) illustrates the network schema for HINs in MOOCs. The schema comprises five distinct entities: course, knowledge concept, video, teacher, and learner. There are various relationships between these entities. For example, learners watch videos, and videos are linked to different knowledge concepts. Based on previous research [4,9], this study also selected the following four meta-paths to represent learners from different dimensions, which are detailed in Table 1.

- $mp_1(U_1 - K - U_2)$ means that two users could make a connection if they have learned the same concept.
- $mp_2(U_1 - C - U_2)$ denotes that users will be connected if they take the same courses.
- $mp_3(U_1 - V - U_2)$ denotes that users who have watched the same video will make a connection.
- $mp_4(U_1 - C_1 - T - C_2 - U_2)$ means that users are connected by taking a course with the same same teacher.

Table 1. Meta-paths selected based on HIN to extract learner–learner and concept–concept relationships.

Type	Meta-Math
Learners	$Learner \rightarrow Concept \xrightarrow{-1} Learner$
	$Learner \rightarrow Course \xrightarrow{-1} Learner$
	$Learner \rightarrow Video \xrightarrow{-1} Learner$
	$Learner \rightarrow Course \rightarrow Teacher \xrightarrow{-1} Course \xrightarrow{-1} Learner.$
Concepts	$Concept \rightarrow Course \xrightarrow{-1} Concept$
	$Concept \rightarrow Learner \xrightarrow{-1} Concept$

Enhancing learner representation with graph-contrastive learning. Recent studies have highlighted the efficacy of graph convolutional networks (GCNs) in representation learning for knowledge-concept recommendations [4,9,10]. To generalize learners’ representations, h_{mp_i} , from an HIN along the each meta-path, GCNs are applied to aggregate the similarity information between learners on different meta-paths with the following layer-wise propagation rule:

$$h_{mp_i}^{l+1} = \sigma(P h_{mp_i}^l W_{mp_i}^l) \tag{1}$$

where $\sigma()$ represents the non-linear activation function, specifically *ReLU*. Matrix P is defined as $D^{-1}\tilde{A}$, with D being the degree matrix of A . $D^{-1}\tilde{A}$ aims to get the average embedding of a node and its neighborhoods. Matrix \tilde{A} is defined as $\tilde{A} = A + I$, where A is the adjacency matrix of the mate-path mp_i , and I is identity matrix. Matrix $W_{mp_i}^{(l)}$ is the trainable weight matrix of the mate-path mp_i at layer l . Here, the mate-path mp_i aggregate the relationships between learners.

As we mentioned before, the neighborhood information of learners has not been explored completely. It is difficult to identify a real neighborhood. We made use of graph-contrastive learning to explore higher-order neighborhood information. The first priority was constructing data augmentation. On each meta-path, the embedding of learners and the embedding of the output of the next layer of GCNs are treated as positive pairs; the propagation rules are shown in Equation (2):

$$h'^{mp_i} = P' h_{mp_i}^l \tag{2}$$

Unlike matrix P defined in Equation (1), weight matrix P' and the nonlinear activation operation are discarded. That is to say that the input only retains the average homogeneous neighborhood embedding.

$h^{l_{mp_i}}$ contains the rich homogeneous graph structural information on the i -th meta-path. Based on the InfoNCE [25], the auxiliary task objectives for each meta-path are described in Equation (3):

$$\mathcal{L}_{self}^{mp_i} = \sum_{p \in P} -\log \frac{\exp((s(\mathbf{H}_p^{(l)}, \mathbf{H}_p^{(l)'})/\tau))}{\sum_{n \in \mathcal{N}_i} \exp((s(\mathbf{H}_p^{(l)}, \mathbf{H}_n^{(l)'})/\tau))} \quad (3)$$

where \mathbf{H}^l and $\mathbf{H}^{l'}$ are the corresponding normalized embedding of the different layers of GCNs. τ is the temperature hyper-parameter of the softmax function. Function $s()$ denotes cosine similarity. \mathcal{N}_i represents the negative sample set of the i -th meta-path, and P is a positive sample set. p can be searched according to hard negative samples. To generate our hard-negative-sample set, we selected samples with similarity values greater than 0 and ranked them from 100 to 200. Additionally, we extracted 10% of the hard negative sample set, while the remaining 90% was obtained through random sampling to serve as negative samples. Of these, 10% were extracted as hard negative samples, while the remaining 90% of the negative samples were obtained through random sampling.

In addition, graph-contrastive learning on different meta-paths has different objectives, which can be formulated as follows:

$$\mathcal{L}_{self} = \sum_{i \in |mp|} \lambda_i \mathcal{L}_{self}^{mp_i} \quad (4)$$

where \mathcal{L}_{self} is the final contrastive loss function, and λ_i represents the different weights of the contrastive loss.

Intuitively, learners' embedding from different meta-paths should contribute different weights to the final results. The importance of meta-paths in learners' final embedding can be formulated as follows:

$$\alpha^{mp_i} = \frac{\exp(\mathbf{V}_e^T \sigma(\mathbf{W}_e \mathbf{h}^{mp_i}))}{\sum_{j \in |mp_c|} \exp(\mathbf{V}_e^T \sigma(\mathbf{W}_e \mathbf{e}^{mp_j}))} \quad (5)$$

The final representation of learners, \mathbf{e} , can be described as follows:

$$\mathbf{e} = \sum_{j \in |mp_e|} \alpha^{mp_j} \mathbf{e}^{mp_j} \quad (6)$$

where mp_e represents the selected meta-paths of learners.

4.2. The Representation Learning of the Concepts

The meta-paths of the concepts are listed in Table 1 as well. They are described as follows:

- $mp_5(K_1 - C - K_2)$ denotes the knowledge concepts that are connected and that have appeared in the same course.
- $mp_6(K_1 - L - K_2)$ indicates that the knowledge concepts were related if they were clicked on by one learner.

Knowledge concepts K embedding x can be learned through a graph heterogeneous attention network to aggregate the information along the selected meta-path. First, graph convolutional networks are utilized to represent the concept based on different meta-paths, which can be formulated as follows:

$$x_{mp_i}^{l+1} = \sigma(Px_{mp_i}^l W_{mp_i}^l) \quad (7)$$

where W^l represents the trainable weight matrix of the meta-path mp_i at layer l . Here, the meta-path mp_i aggregates the relationships between concepts.

Following [10], the representations with different meta-paths give different contributions to the final results. The importance of different meta-paths can be obtained via Equation (5):

$$\alpha^{mp_i} = \frac{\exp(\mathbf{V}^T \sigma(\mathbf{W}_c \mathbf{x}^{mp_i}))}{\sum_{j \in |mp_c|} \exp(\mathbf{V}^T \sigma(\mathbf{W}_c \mathbf{x}^{mp_j}))} \tag{8}$$

where \mathbf{V}^T and \mathbf{W}_c are the trainable weight matrixes. σ is the tanh function. The final representation of concepts can be described as follows:

$$\mathbf{z} = \sum_{j \in |mp_z|} \alpha^{mp_j} \mathbf{x}^{mp_j} \tag{9}$$

where mp_z represents the selected meta-paths of concepts. With the final representations, we make use of the inner product to predict how likely a learner, u , would be to click on the concept k :

$$\hat{y}_{u,k} = e_u^T z_k \tag{10}$$

Here, e_u and z_k are normalized final representations of u and k that belong to e and z .

4.3. Multi-Task Training

A multi-task strategy is used to jointly optimize the main supervised task and the auxiliary graph-contrastive learning tasks. Specifically, the joint loss can be shown as follows:

$$\mathcal{L} = \mathcal{L}_{main} + \mu \mathcal{L}_{self} + \lambda \|\Theta\|^2 \tag{11}$$

where μ is the regularization strength of the contrastive learning tasks, and λ is the regularization strength of the L2 normalization.

Loss for the main task. A different main loss could be chosen according to the objectives. In knowledge-concept recommendations, we find that there exist two tendencies while learners are learning related concepts. First, learners from different professional backgrounds, aside from the knowledge concepts of specific general education courses, exhibit differences between groups divided by their professional backgrounds. Second, as they delve deeper into their studies, similar learners will become interested in the same concepts, showing high correlations within the group. To encourage the above properties, InfoNCE [26] loss is adopted, and it could maximize the mutual information of the positive-sample pair and minimize the mutual information of the negative-sample pair. The main loss function can be described as follows:

$$\mathcal{L}_{main} = \sum_{u,i \in UK} -\log \frac{\exp((s(\mathbf{C}Z_u, \mathbf{X}_i)/\tau))}{\sum_{j \in N} \exp((s(\mathbf{Z}_u, \mathbf{X}_j)/\tau))} \tag{12}$$

where Z_u is the user embedding, and u comes from learners via random sampling. X_j is the embedding of a concept. X_j is the concept embedding obtained via global uniform random sampling from the knowledge concept set. The algorithm of KCRHGCL is detailed in Algorithm 1.

Algorithm 1: KCRHGCL

Input: Given the meta-paths \mathcal{MP} , the interactive data matrix \mathcal{R} , the global epoch E , the batch size N , the number of negative samples in main tasks n_1 , and the number of negative samples in auxiliary tasks n_2

Output: the list of predicted knowledge concepts *list*

- 1 Initialize learner embedding h_0 and concept embedding x_0 ;
- 2 **foreach** mp in \mathcal{MP} **do**
- 3 | $P = D^{-1}(A + I)$
- 4 **end**
- 5 **for** $episode = 1 \dots E$ **do**
- 6 | **foreach** mp in $\mathcal{MP}_{\mathcal{L}}$ **do**
- 7 | Compute the representation on different meta-path of learners h_{mp}^{l+1}
- 7 | Compute the representation on different meta-path of learners via graph-contrastive learning $h_{mp}^{l+1'}$
- 8 | **end**
- 9 | **foreach** mp in $\mathcal{MP}_{\mathcal{K}}$ **do**
- 10 | Compute the representation on different meta-path of concepts x_{mp}^{l+1}
- 11 | **end**
- 12 | Recompute learner embedding e and concept embedding z via attention mechanisms Compute the prediction $\hat{Y} = e^T z$ Compute the loss $L = L_{main} + \mu L_{self} + \beta L_2$ Calculate gradient $\nabla_{\theta} L(f_{\theta})$ Update θ with $\theta \leftarrow \theta - \eta \nabla_{\theta} L(f_{\theta})$
- 13 **end**

5. Experiments

This section describes experiments that were conducted on a real-world dataset named MOOCCube to evaluate the effectiveness of KCRHGCL for recommending knowledge concept tasks. To achieve this, the following research questions were addressed:

- RQ1: How effective is KCRHGCL in the concept recommendation compared with other baselines?
- RQ2: How does the performance of KCRHGCL vary with different meta-paths?
- RQ3: How do the different modules in KCRHGCL, such as using graph structure-aware contrastive learning on meta-paths to enhance the structural features of learner groups, or using InfoNCE as the main function to increase inter-group structural differences, contribute to the overall performance of the method?
- RQ4: How does the performance of KCRHGCL vary with different hyper-parameter settings, such as the embedding dimension size, the regularization weights, and the layer number of graph-based message propagation?
- RQ5: How does the convergence of the KCRHGCL method compare to the suboptimal method MOOCIR?

5.1. Experimental Settings

5.1.1. Datasets

Experiments were organized on the MOOCCube dataset [5] to validate the effectiveness of the method. MOOCCube is a large-scale, high-quality dataset designed for AI research in the field of education. Table 2 provides a detailed overview of the MOOCCube dataset used in our experiments, and it can be observed that the interactions between entities are sparse. The data preprocessing approach we adopted is consistent with that of [9]. Specifically, we extracted data from 1 January 2017 to 31 October 2019 for training and data from 1 November 2019 to 31 December 2019 for testing. Furthermore, we ensured that the data met two criteria: (i) learners were present in both the training and test datasets, and (ii) at least one new concept appeared in the test data.

Table 2. Details of the MOOCCube dataset.

Entities	Statistics	Relations	Statistics
Users	2005	User-concept	930,553
Concepts	21,037	user-course	13,696
Courses	600	Course-video	42,117
Videos	22,403	Teacher-course	1875
Schools	137	Video-concept	295,475
Teachers	138	Course-concept	150,811

5.1.2. Evaluation Metrics

Several evaluation metrics were selected in this paper, and they are commonly used in recommendation tasks. The hit ratio of the top K ($HR@K$) is a measure of recall that indicates the percentage of relevant concepts in the test set that appear in the top- K recommended concepts. It is a widely used measure of recall, and it assesses a system's ability to retrieve relevant concepts [27]. The normalized discounted cumulative gain of the top K ($NDCG@K$) is a ranking metric that takes into account the relevance of concepts in the top- K recommended items, as well as their positions in the ranking. It is a normalized measure of the cumulative graded relevance based on the top- K recommended concepts for all learners of recommendations [28]. The mean reciprocal rank (MRR) was used to evaluate the ranking of search results. It measures the effectiveness of the ranking by calculating the average reciprocal rank of relevant concepts in the ranking [29].

5.1.3. Compared Methods

To evaluate the validity of KCRHGCL, we compared KCRHGCL with the following state-of-the-art methods that utilized various deep-learning methods to improve the performance of recommendation tasks. The baseline details are described below.

- FISM [30] is an item-to-item collaborative-filtering recommendation method that predicts user preferences by aggregating the latent factors of the items the user has interacted with.
- NAIS [31] is another item-to-item collaborative filtering recommendation method that introduces an attention mechanism to weigh the importance of different historical interactions for a user. Therefore, NAIS focuses more on the interactions that are more relevant to predicting the user's preference for a specific item.
- Metapath2vec [32] is a representation learning method based on a heterogeneous information network. Random walk and skip-gram are two key components. By maximizing the co-occurrence probability of nodes and their neighboring nodes in random walk sequences, Metapath2vec learns high-quality node representations that capture the structural and semantic relationships within a network.
- ACKRec [4] is an end-to-end graph neural network-based method. It operates on heterogeneous graphs, defines predefined metapaths, uses graph convolution to extract entity representations from different metapaths, and employs an attention mechanism to aggregate the representations of different metapaths.
- MOOCIR [9] is another end-to-end graph neural network-based method for recommendations. Unlike ACKRec [4], MOOCIR uses implicit feedback instead of ratings and extends matrix factorization by incorporating learner and concept representations learned from different metapaths.
- HinCRec-RL [10] is a recommendation method that combines heterogeneous information networks and reinforcement learning for KCR to enhance performance.

5.1.4. Implementation Details

To ensure fairness, all of our methods were implemented using Python and run on a Linux server with two 2080ti GPUs and 64 GB of memory. The embedding initialization was obtained from the data provided by [9]. Our methods were optimized using the Adam

optimizer [33], with an initial learning rate of 0.01. In addition, we used a single graph-based message propagation layer to avoid overfitting, and a dropout rate of 0.2 was applied. To reduce the impact of random numbers on the experimental results, we took the average of multiple operations. It is worth noting that the results of the compared methods were mostly from [9].

5.2. Performance Comparison

Table 3 shows the performance of different methods using the MOOCCube dataset. From the look of the results, the following conclusions can be drawn:

- It is evident that all the methods except for FISM and NAIS were based on an HIN. The methods based on HIN outperformed those that were not based on an HIN. For instance, metapath2vec outperformed FISM in terms of $HR@5$, $NDCG@5$, and MRR by over 9.9%, 6.9%, 5.3%, respectively. This indicates that constructing an HIN is an effective approach to alleviating sparsity in KCR tasks.
- KCRHGCL outperformed all other state-of-the-art recommendation methods with the MOOCCube dataset, with significant improvements in the $HR@5$, $NDCG@5$, and MRR metrics of 14.0%, 12.5%, 10.7%, respectively, when compared to ACKRec. This remarkable performance can be attributed to two key factors. Firstly, KCRHGCL incorporates self-supervised signals from different views to capture learners' interests by identifying structured similar neighbors. Secondly, the InfoNCE loss was utilized to adapt to the objective that conforms to the characteristics of learners' learning behavior.

Table 3. The performance of different methods with the MOOCCube dataset was compared, and the best-performing scores are marked in bold.

Methods	HR@5	HR@10	HR@20	NDCG@5	NDCG@10	NDCG@20	MRR
FISM	0.584	0.701	0.800	0.438	0.476	0.501	0.418
NAIS	0.568	0.691	0.811	0.420	0.461	0.491	0.403
metapath2vec	0.642	0.773	0.873	0.468	0.511	0.537	0.440
ACKRec	0.659	0.764	0.842	0.503	0.538	0.557	0.475
HinCRec-RL	0.662	0.815	0.887	0.462	0.513	0.532	0.428
MOOCIR	0.704	0.836	0.922	0.520	0.562	0.584	0.484
KCRHGCL	0.751	0.866	0.934	0.566	0.603	0.621	0.526

5.3. Effect of Different Meta-Paths

In this section, we explore the impact of different meta-paths and their combinations on the performance of our proposed method. We selected four meta-paths to represent the relationships between learners: $mp_1 : U_1 - K - U_2$, $mp_2 : U_1 - C - U_2$, $MP3 : U_1 - V - U_2$, and $MP4 : U_1 - C_1 - T - C_2 - T_2$. We present the results of different combinations of meta-paths in Table 3.

Interestingly, we found that selecting only a single meta-path, $MP2$, yielded the best performance, which contrasts with the findings of [4], where using more meta-paths led to better results. One possible explanation for this discrepancy is that, in MOOCs, knowledge concepts are structured around courses, and most learners acquire knowledge concepts centered around courses. Therefore, self-supervision signals that capture this feature more comprehensively may lead to better performance.

Furthermore, we observe that each individual meta-path produces different performance, with the order being $MP2 > MP4 > MP3 > MP1$. Additionally, the results of different combinations of meta-paths fall between the results of the individual meta-paths.

5.4. Ablation Study

This section proposes three variants of the KCRHGCL. Each variant focuses on a specific technique to improve the method's effectiveness. The three variants are as follows:

- $KCRHGCL_{nos}$: This method employs a single meta-path (mp_2) to learn node embedding, and it uses Bayesian personalized ranking (BPR) as the primary loss function. It does not utilize any additional structural neighbor information.
- $KCRHGCL_s$: This variant introduces structural neighbor information to explore the high-order similarity of learners on different meta-paths. BPR is still used as the main loss function.
- $KCRHGCL_i$: In this method, InfoNCE loss is used as the primary training objective. The approach aims to keep concepts that learners are not interested in away and bring concepts that learners are interested in closer.

The performance of three variants of KCRHGCL is shown in Table 4. The results in Table 5 indicate that utilizing the InfoNCE loss function leads to a significant improvement in performance for the HR@5, NDCG@5, and MRR metrics compared to using the BPR loss function. Comparing $KCRHGCL_{nos}$ to $KCRHGCL_s$ reveals that it improves HR@5, NDCG@5, MRR over 22.8%, 31.5%, 30.9%. This suggests that InfoNCE loss is a more suitable training objective for KCR tasks than BPR, as it better distinguishes the interests and preferences of different professional learners, and it can better capture the similarity between learners with similar interests. When comparing $KCRHGCL$ to $KCRHGCL_i$, there is a small improvement in performance for the HR@5, NDCG@5, and MRR metrics of approximately 0.9%, 1.1%, 1.0%, respectively. This suggests that the enhanced structural neighbor achieved through graph-contrastive learning in the meta-path can contribute to the final performance of the model.

Table 4. Results of different combinations of meta-paths with the best-performing scores marked in bold.

Meta-Path	HR@5	HR@10	HR@20	NDCG@5	NDCG@10	NDCG@20	MRR
mp_1	0.713	0.840	0.918	0.533	0.574	0.594	0.498
mp_2	0.751	0.866	0.934	0.566	0.603	0.621	0.526
mp_3	0.718	0.840	0.919	0.536	0.576	0.596	0.500
mp_4	0.729	0.852	0.925	0.542	0.582	0.600	0.503
$mp_1 \& mp_2$	0.726	0.845	0.920	0.543	0.582	0.595	0.506
$mp_1 \& mp_3$	0.717	0.844	0.924	0.533	0.575	0.595	0.497
$mp_1 \& mp_4$	0.718	0.844	0.924	0.534	0.575	0.596	0.498
$mp_2 \& mp_4$	0.723	0.847	0.924	0.541	0.581	0.601	0.505
$mp_2 \& mp_3$	0.738	0.854	0.926	0.555	0.593	0.611	0.517
$mp_3 \& mp_4$	0.724	0.848	0.925	0.542	0.582	0.602	0.506
$mp_1 \& mp_2 \& mp_3$	0.720	0.843	0.921	0.538	0.579	0.598	0.502
$mp_1 \& mp_2 \& mp_4$	0.723	0.847	0.925	0.538	0.579	0.599	0.502
$mp_1 \& mp_3 \& mp_4$	0.718	0.841	0.920	0.536	0.577	0.597	0.501
$mp_2 \& mp_3 \& mp_4$	0.728	0.849	0.925	0.545	0.584	0.604	0.508
All	0.726	0.845	0.921	0.543	0.581	0.601	0.505

Table 5. Performance results of three variants of KCRHGCL with the best scores are marked in bold.

Methods	HR@5	HR@10	HR@20	NDCG@5	NDCG@10	NDCG@20	MRR
$KCRHGCL_{nos}$	0.584	0.742	0.854	0.394	0.446	0.474	0.365
$KCRHGCL_s$	0.606	0.755	0.859	0.426	0.475	0.501	0.398
$KCRHGCL_i$	0.744	0.860	0.930	0.560	0.597	0.615	0.521
$KCRHGCL$	0.751	0.866	0.934	0.566	0.603	0.621	0.526

5.5. Parameter Analysis

In this section, we present the results of further conducted experiments to show the effects of the three following hyper-parameters: (1) the embedding dimension size d ; (2) the regularization weights μ in Equation (11); (3) the layer number of graph-based message propagation l . The experimental results are represented in Table 6 and Figures 3 and 4.

Table 6. The impact of the layer number of graph-based message propagation l , the best scores are marked in bold.

Layers	HR@5	HR@10	HR@20	NDCG@5	NDCG@10	NDCG@20
1	0.7514	0.8663	0.9345	0.5658	0.6033	0.6207
2	0.7515	0.8676	0.9337	0.5646	0.6025	0.6195
3	0.7511	0.8665	0.9326	0.5651	0.6028	0.6198

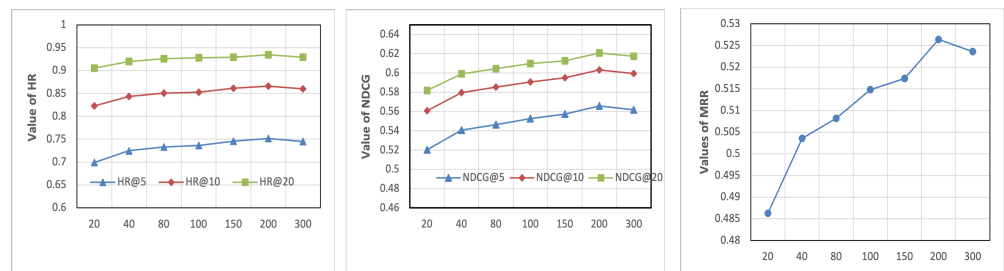


Figure 3. The results of the embedding dimension size d .

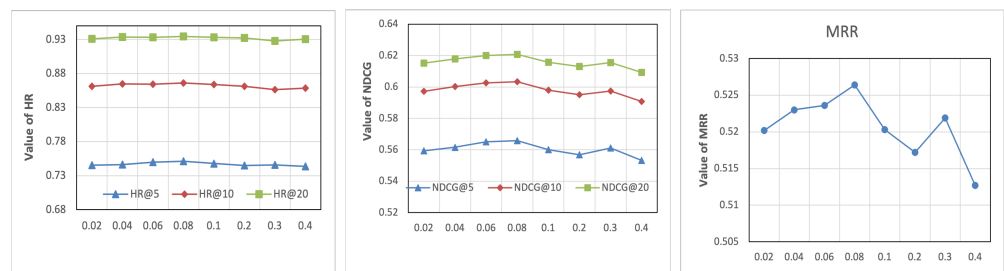


Figure 4. The results of the regularization weights μ .

The embedding dimension size d plays an important role in the method’s performance. It is important to note that increasing the embedding dimension size d can also result in increased computational complexity and memory usage. Therefore, the choice of d should be balanced between performance and efficiency. It was observed that $HR@5$, $HR@10$, and $HR@20$ all improve as dimension d increases, but the rate of improvement gradually slows down. At lower dimensions (such as $d = 20$), HR increases quickly, whereas at higher dimensions (such as $d \geq 200$), the improvement in HR tends to plateau or even show no change. This indicates that, beyond a certain dimension, increasing the dimension has a limited effect on improving HR, and it may even lead to model overfitting. $NDCG@5$, $NDCG@10$, $NDCG@20$, and MRR follow the same trend, indicating that lower dimensions may not fully capture the underlying structure of the data. Increasing the dimensions appropriately can improve the model performance, but excessively high dimensions may increase the model’s complexity, leading to a slowdown in the performance improvement.

The regularization weights μ are introduced to balance the main and auxiliary losses, and we experimented with its value within the range of $[0.02, 0.04, 0.06, 0.08, 0.1, 0.2, 0.3, 0.4]$. The results are reported in Figure 3. Our findings show that an appropriate μ value can lead to better performance for KCRHGCL. Specifically, the best performance was achieved with $\mu = 0.08$.

The layer number of graph-based message propagation l was searched from 1 to 3 in GCNs. The results in Table 6 show that the methods achieve better performance with one or two layers for all the metrics. It is possible that lower layers can capture sufficient features, while more layers can lead to over-smoothing and introduce noise to the feature representation.

5.6. Convergence Analysis

The convergence rate analysis presented in Figure 5 demonstrates that the proposed method achieves a stable state after nine epochs of training. This suggests that the model has converged to a reliable solution within a relatively small number of epochs. In comparison, the MOOCIR model reaches a stable state after 15 epochs, indicating a slower convergence rate. Overall, the results suggest that the proposed method has a more robust convergence rate than the MOOCIR model.

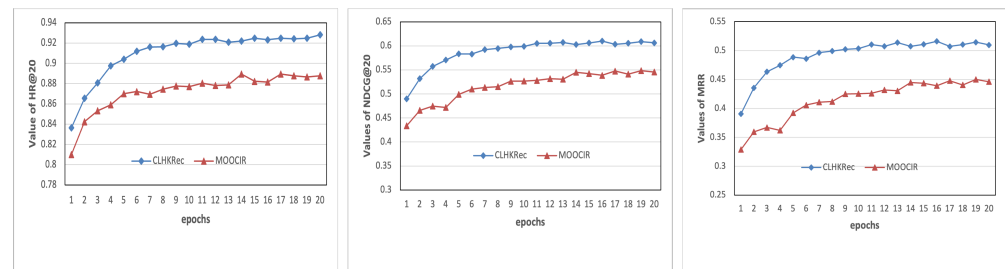


Figure 5. Convergence analysis for KCRHGCL.

5.7. Experimental Results

Extensive experiments were organized on MOOCube to verify the effectiveness of the KCRHGCL method. Compared to other state-of-the-art methods, KCRHGCL showed improvements of 6.68%, 8.85%, and 8.68% in $HR@5$, $NDCG@5$, and MRR , respectively, over the suboptimal methods. The impact of different meta-path choices on performance was analyzed, revealing that, due to the course-centric organization of existing MOOCs, the $L_1 - C - L_2$ meta-path yielded better results. Ablation experiments validated the necessity of using heterogeneous graph-contrastive learning to enhance learner representations by exploring the similarity of structural neighbors in different dimensions, as well as the importance of negative sampling and contrastive loss functions in distinguishing structural differences. The study examined the impact of hyperparameters on KCRHGCL, identifying the optimal hyperparameters. A comparison and analysis of the convergence characteristics of KCRHGCL and MOOCIR showed that KCRHGCL converges faster.

6. Conclusions

In this paper, a multi-task knowledge-concept recommendation method called Contrastive Learning from Heterogeneous Information Networks has been proposed. The goal of KCRHGCL is to capture the group structural characteristics of learners' preferences as much as possible. On the one hand, the similarity within the structure of learner groups was further explored to enhance learner representations via heterogeneous graph-contrastive learning; on the other hand, hard negative sampling techniques and contrastive loss functions were used to guide the differentiation of learners between different group structures, thereby improving the recommendation effectiveness. Extensive experiments were organized on MOOCube to verify the effectiveness of the KCRHGCL method. Compared to other state-of-the-art methods, KCRHGCL showed improvements of 6.68%, 8.85%, and 8.68% in $HR@5$, $NDCG@5$, and MRR , respectively. The performance of KCRHGCL is indeed impressive. However, it is worth noting that learners' study interests may change over time, whether in the short term or the long term. Therefore, it would be intriguing to explore methods that can capture such shifts in knowledge-concept recommendations in future research.

Author Contributions: Methodology, Y.L.; validation, W.W.; resources and Writing—original draft, L.W.; writing—review and editing, Y.Z. All authors have read and agreed to the published version of the manuscript.

Funding: This research was funded by the Pedagogy Program “The 13th Five Year Plan” of the National Social Science Foundation in 2019, grant No. BBA190021.

Data Availability Statement: The original contributions presented in the study are included in the article, further inquiries can be directed to the corresponding author.

Conflicts of Interest: The authors declare no conflicts of interest. The funders had no role in the design of the study, in the collection, analyses, or interpretation of data, in the writing of the manuscript, or in the decision to publish the results.

References

1. Adeniyi, I.S.; Al Hamad, N.M.; Adewusi, O.E.; Unachukwu, C.C.; Osawaru, B.; Chilson, O.U.; Omolawal, S.A.; Aliu, A.O.; David, I.O. Reviewing online learning effectiveness during the COVID-19 pandemic: A global perspective. *Int. J. Sci. Res. Arch.* **2024**, *11*, 1676–1685. [\[CrossRef\]](#)
2. Xia, D.; Zhang, Y.; Qiu, Y.; Zhang, S.; Tian, Y.; Zhao, X. Research on the Dynamic Evolution Law of Online Knowledge Sharing Under Trust. *Int. J. Chang. Educ.* **2024**, *1*, 32–40. [\[CrossRef\]](#)
3. Pan, L.; Wang, X.; Li, C.; Li, J.; Tang, J. Course concept extraction in moocs via embedding-based graph propagation. In Proceedings of the Eighth International Joint Conference on Natural Language Processing (Volume 1: Long Papers), Taipei, Taiwan, 27 November–1 December 2017; pp. 875–884.
4. Gong, J.; Wang, S.; Wang, J.; Feng, W.; Peng, H.; Tang, J.; Yu, P.S. 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.
5. Yu, J.; Luo, G.; Xiao, T.; Zhong, Q.; Wang, Y.; Feng, W.; Luo, J.; Wang, C.; Hou, L.; Li, J.; et al. MOOCube: 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.
6. Tang, C.L.; Liao, J.; Wang, H.C.; Sung, C.Y.; Lin, W.C. Conceptguide: Supporting online video learning with concept map-based recommendation of learning path. *Proc. Web Conf.* **2021**, *2021*, 2757–2768.
7. Zhang, J.; Hao, B.; Chen, B.; Li, C.; Chen, H.; Sun, J. Hierarchical reinforcement learning for course recommendation in MOOCs. In Proceedings of the AAAI Conference on Artificial Intelligence, Honolulu, HI, USA, 27 January–1 February 2019; Volume 33, pp. 435–442.
8. Al-Twijri, M.I.; Luna, J.M.; Herrera, F.; Ventura, S. Course Recommendation based on Sequences: An Evolutionary Search of Emerging Sequential Patterns. *Cogn. Comput.* **2022**, *14*, 1474–1495. [\[CrossRef\]](#)
9. Piao, G. Recommending Knowledge Concepts on MOOC Platforms with Meta-Path-Based Representation Learning. In *International Educational Data Mining Society, Proceedings of the International Conference on Educational Data Mining (EDM), Online, 29 June–2 July 2021*; International Educational Data Mining Society: Worcester MA, USA, 2021.
10. 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.
11. Lin, Y.; Lin, F.; Yang, L.; Zeng, W.; Liu, Y.; Wu, P. Context-aware reinforcement learning for course recommendation. *Appl. Soft Comput.* **2022**, *125*, 109189. [\[CrossRef\]](#)
12. 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. [\[CrossRef\]](#)
13. Jing, X.; Tang, J. Guess you like: Course recommendation in MOOCs. In Proceedings of the International Conference on Web Intelligence, Leipzig, Germany, 23–26 August 2017; pp. 783–789.
14. Jung, H.; Jang, Y.; Kim, S.; Kim, H. KPCR: Knowledge graph enhanced personalized course recommendation. In Proceedings of the AI 2021: Advances in Artificial Intelligence: 34th Australasian Joint Conference, AI 2021, Sydney, Australia, 2–4 February 2022; Springer: Berlin/Heidelberg, Germany, 2022; pp. 739–750.
15. Liu, X.; Zhang, F.; Hou, Z.; Mian, L.; Wang, Z.; Zhang, J.; Tang, J. Self-supervised learning: Generative or contrastive. *IEEE Trans. Knowl. Data Eng.* **2021**, *35*, 857–876. [\[CrossRef\]](#)
16. Wu, J.; Wang, X.; Feng, F.; He, X.; Chen, L.; Lian, J.; Xie, X. Self-supervised graph learning for recommendation. In Proceedings of the 44th International ACM SIGIR Conference on Research and Development in Information Retrieval, Virtual, 11–15 July 2021; pp. 726–735.
17. Jing, M.; Zhu, Y.; Zang, T.; Wang, K. Contrastive self-supervised learning in recommender systems: A survey. *ACM Trans. Inf. Syst.* **2023**, *42*, 1–39. [\[CrossRef\]](#)
18. Xie, X.; Sun, F.; Liu, Z.; Wu, S.; Gao, J.; Ding, B.; Cui, B. Contrastive learning for sequential recommendation. *arXiv* **2020**, arXiv:2010.14395.
19. Yang, Y.; Huang, C.; Xia, L.; Li, C. Knowledge graph contrastive learning for recommendation. In Proceedings of the 45th International ACM SIGIR Conference on Research and Development in Information Retrieval, Madrid, Spain, 11–15 July 2022; pp. 1434–1443.
20. Cai, X.; Huang, C.; Xia, L.; Ren, X. LightGCL: Simple yet effective graph contrastive learning for recommendation. *arXiv* **2023**, arXiv:2302.08191.
21. Lin, Z.; Tian, C.; Hou, Y.; Zhao, W.X. Improving Graph Collaborative Filtering with Neighborhood-enriched Contrastive Learning. In Proceedings of the ACM Web Conference 2022, Lyon, France, 25–29 April 2022; pp. 2320–2329.

22. Chen, M.; Huang, C.; Xia, L.; Wei, W.; Xu, Y.; Luo, R. Heterogeneous graph contrastive learning for recommendation. In Proceedings of the Sixteenth ACM International Conference on Web Search and Data Mining, Singapore, 27 February–3 March 2023; pp. 544–552.
23. Zheng, X.; Wu, R.; Han, Z.; Chen, C.; Chen, L.; Han, B. Heterogeneous information crossing on graphs for session-based recommender systems. *ACM Trans. Web* **2024**, *18*, 1–24. [[CrossRef](#)]
24. Sun, Y.; Han, J.; Yan, X.; Yu, P.S.; Wu, T. Pathsim: Meta path-based top-k similarity search in heterogeneous information networks. *Proc. VLDB Endow.* **2011**, *4*, 992–1003. [[CrossRef](#)]
25. Wu, C.; Wu, F.; Huang, Y. Rethinking infonce: How many negative samples do you need? *arXiv* **2021**, arXiv:2105.13003.
26. Oord, A.v.d.; Li, Y.; Vinyals, O. Representation learning with contrastive predictive coding. *arXiv* **2018**, arXiv:1807.03748.
27. Cremonesi, P.; Koren, Y.; Turrin, R. Performance of recommender algorithms on top-n recommendation tasks. In Proceedings of the Fourth ACM Conference on Recommender Systems, Barcelona, Spain, 26–30 September 2010; pp. 39–46.
28. Wang, Y.; Wang, L.; Li, Y.; He, D.; Chen, W.; Liu, T. A theoretical analysis of normalized discounted cumulative gain (NDCG) ranking measures. In Proceedings of the 26th Annual Conference on Learning Theory (COLT 2013), Princeton, NJ, USA, 12–14 June 2013; Citeseer: Camp Hill, PA, USA, 2023.
29. Lu, X.; Wu, J.; Yuan, J. Optimizing Reciprocal Rank with Bayesian Average for improved Next Item Recommendation. In Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, Taipei, Taiwan, 23–27 July 2023; pp. 2236–2240.
30. Kabbur, S.; Ning, X.; Karypis, G. Fism: Factored item similarity models for top-n recommender systems. In Proceedings of the 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Chicago, IL, USA, 11–14 August 2013; pp. 659–667.
31. He, X.; He, Z.; Song, J.; Liu, Z.; Jiang, Y.G.; Chua, T.S. Nais: Neural attentive item similarity model for recommendation. *IEEE Trans. Knowl. Data Eng.* **2018**, *30*, 2354–2366. [[CrossRef](#)]
32. Dong, Y.; Chawla, N.V.; Swami, A. metapath2vec: Scalable representation learning for heterogeneous networks. In Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, Halifax, NS, Canada, 13–17 August 2017; pp. 135–144.
33. Kingma, D.P. A method for stochastic optimization. In Proceedings of the 3rd International Conference on Learning Representations, ICLR, San Diego, CA, USA, 7–9 May 2015.

Disclaimer/Publisher’s Note: The statements, opinions and data contained in all publications are solely those of the individual author(s) and contributor(s) and not of MDPI and/or the editor(s). MDPI and/or the editor(s) disclaim responsibility for any injury to people or property resulting from any ideas, methods, instructions or products referred to in the content.