Reinforced Concept Recommendation in MOOCs with Heterogeneous Information Networks Learning

***Abstract*—In MOOCs, commonly speaking, curriculum designing, course selection and knowledge concept recommendation are the three major steps that systematically instruct users to learn. This paper focuses on the knowledge concept recommendation in MOOCs, which strives to recommend related topics to users to facilitate their online study. Despite the existing efforts in some related areas such as course recommendation, there are still two main issues hindering them from being extented to accurately recommend concept to users that with limited information. First, the existing approaches only consider the historical behabiour of users, but ignoring various kinds of auxiliary data (a.k.a., side information), which is also critical for better user embedding. Second, traditional recommendation models only consider the immediate user response on the recommended item, not explicitly taking into account the long-term user interest.**

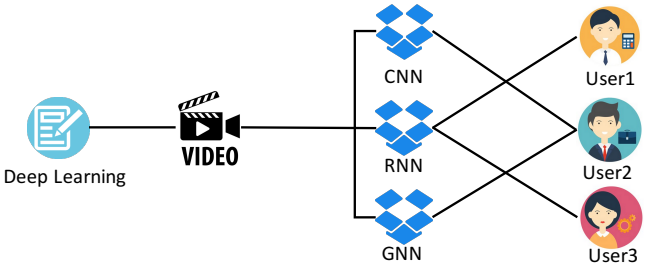
To tackle the two aforementioned issues, this paper proposes HIN-RL, a novel reinforced concept recommendation model in MOOCs with heterogeneous information networks learning. We first formulate the concept recommendation in MOOCs as a reinforcement learning problem to offer personalized and dynamic knowledge concept label list to users/students. To consider more auxiliary information of users, we construct a heterogeneous information network among user, course, concept, and then exploit meta-path to aggregate information from neighbours for better representation. Comprehensive experiments and analysis on a large-scale real-world dataset collected from *XuetangX*, one of the largest MOOCs in China, show that our proposed model can accurately recommend concept to users and outperforms some state-of-the-art methods.

***Index Terms*—Concept recommendation, MOOCs, Heterogeneous Information Network, Reinforcement learning, Meta-path**

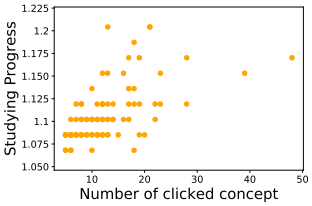
I. Introduction

Massive Open Online Courses (MOOCs), which aim at providing a large-scale interactive participation and open access via the web, are rapidly becoming an established method of online and distance education. *XuetangX* MOOCs, as one of the largest MOOCs platforms in China, has offered more than 1,000 courses and attracted over 6,000,000 users across the world. In MOOCs, we use *course concepts* to refer to the knowledge concepts taught in the course videos, and related topics that help students better understand course videos. The goal of concept recommendation is to recommender related topics to users to facilitate their online study.

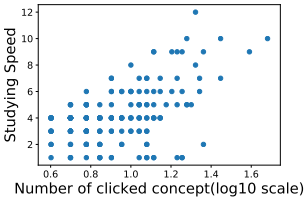
Existing Efforts and Limitations. Many existing efforts have been made towards user behaviour understanding and concept extraction in MOOCs, such as prerequisite relation mining between knowledge concepts [1], course concept extraction [2], learning behavior predicting [3], course recommendation [4] [5], etc. Pan et al. propose to learn latent representations via an embedding-based method for course concept extraction in MOOCs [2]. Both user interests and profiles are employed to feed a proposed content-aware deep learning framework for course recommendation in MOOCs [4] [5].



(a) The network among course, video, concept and users/students in MOOCs.



(b) Studying progress



(c) Studying speed

Fig. 1: **An illustration of concept recommendation.** For this study, we formally define two metrics, *Studying Progress* (*SP*) and *Studying Speed* (*SS*), in order to measure the effectiveness and efficiency of a user on learning a course in MOOCs, respectively. Here, $SP = N_v/N_c$ where N_v denotes the number of videos watched by the user and N_c states the total number of videos contained by the course. $SS = N_f/N_d$ where N_f defines the number of videos within N_d weeks after the user begin to watch the first video of the course. We can find such a significant fact that the more concepts of a course a user learns, the better effectiveness and efficiency of the course he/she will achieve.

However, these approaches still suffer from two major limitations: (a) they ignore rich heterogeneous information across MOOCs. The approach [5] fully considers the semantic information of user profiles and leverage diverse historical courses to make personalized course recommendation. But it is not enough in mining potential semantics because more abundant semantic information hidden in relations between different entities in MOOCs isn't exploited. (b) they cannot consider current reward and future reward simultaneously from online recommendation in the dynamic learning environment. For example, although Pan et al. leverage the demographics

and course prerequisite relation to better reveal users potential choice, they overlook the MOOCS system as a dynamic learning environment, unable to model the current reward and future reward of users' choice [2]. This results in its approach cannot provide personalized candidate concepts.

Insights. These aforementioned limitations motivate us to design a model which learns a more comprehensive representation on users as well as with the ability of personalized concept recommendation. To the best of our knowledge, we are the first to model *XuetangX* MOOCs as a Heterogeneous Information Network (HIN) and propose a meta-path based concept recommendation framework with Reinforcement Learning (RL) to offer personalized and dynamic knowledge concept label list to students for getting a course certificate. For limitation (a), it is a common sense that the students have very diverse backgrounds or levels; and progress conducted in MOOCs is well gained by certain students, but might be hard to others. Meanwhile, existing concept labels listed below in a course video was previously provided by the teacher, but only at a fixed and unpersonalized way, namely, it neither dynamically considers users historical learning behaviors, or utilize the other users' global progress across the MOOCs. For limiation (b), The reasons why we employed reinforcement learning in this study include: 1) The concept clicking rate in MOOCs is fairly sparse [6], 2) in real online learning scenarios of MOOCs, the recommender usually interacts with the user for multiple rounds [7], and 3) interactions between students and recommender agent should be sequential [8].

Fig. 1 illustrates our motivation of this work. Fig. 1a gives an example to show the relationships among course, video, concept and users/students in MOOCs. Specifically, a course *Deep Learning* may contain many related videos and each of them also includes many concepts labels, e.g., *CNN, RNN and GNN*, and different users can click many concept labels to instruct his/her learning of *Deep Learning* course. Fig. 1b tells us that the more concept labels are clicked by a user when learning a course, the bigger the *Studying Process* will be achieved by him/her. Fig. 1c states that the more concept labels are clicked by a user when learning a course, the faster the *Studying Speed* will be obtained by him/her. Both of them deliver the significant importance of concepts on users' learning in MOOCs. The former *studying progress* denotes the extent to which the concept label clicking can influence the effectiveness online course learning of a user, but the latter *studying speed* states to what extent the course can be learned by a user considering the efficiency.

Challenges. The challenges of our study are three-fold: (a) Meta-path based spatial semantic information (e.g., heterogeneous network structure) is intangible; user historic learning behaviors (e.g., historic course/concept click) is temporal and enormous; entity attribute information (e.g., profile information and concept relations) is diverse. Thus, how to intelligently combine these features to design a unified and personalized knowledge concept recommendation framework is a great challenge. (b) In MOOCs, the acquisition of new observations may not be possible anymore; the "environment changing" issue exists when considering dynamic action space (user and concept space) and thus the state transition probabilities will vary. Therefore, how to construct a reward function is another challenge when we apply deep reinforcement learning to the studied problem. (c) The heterogeneous network structure in MOOCs contains rich and irreplaceable information, but also potentially incompatible semantics. Further, how to embed the network structure into low-dimensional intensive features for concept recommendation is another issue, which poses a special challenge to embedding learning in MOOCs.

Our Solutions and Contributions. To tackle the two aforementioned limitations, in this paper, we propose a novel model called Heterogeneous Information Networks with Reinforcement Learning (HIN-RL) for concept recommendation in MOOCs. To address the aforementioned challenges, a potential solution is to utilize the techniques of RL to model the multi-round recommendations as a multi-step decision making problem of concept learning in MOOCs. We are also aware of the importance of semantic recommendation for knowledge concept in rich heterogeneous information network of *XuetangX* MOOCs. Thus, we propose a meta-path based concept recommendation framework with RL, to offer personalized and dynamic knowledge concept label list to students. To the best of our knowledge, concepts recommendation for instructing user to learn at a more fine-grained way has not been studied yet. In MOOCs, a concept is usually considered to be a more fine-grained learning unit than a courses.

In summary, the main contributions of this paper are as follows.

- Unlike the previous studies, we investigate the problem of concept recommendation, a more fine-grained than course recommendation, in *XuetangX* MOOCs from the perspective of reinforcement learning framework.
- We propose a meta-path based user embedding approach with hierarchical attention networks to better represent users in the HIN of MOOCs, and we incorporate it into the reinforcement learning framework.
- To verify the effectiveness of our proposed model, we validate our proposed model on a real-world dataset collected from *XuetangX* MOOCs. Comprehensive experiments and analysis show the effectiveness of our proposed model when compared with some state-of-the-art methods.

II. Preliminaries

Before we introduce our proposed method, in this section, we first introduce some background about HIN and the RL-based framework. Furthermore, we will give some related preliminaries about our proposed method.

A. Heterogeneous Information Network

A HIN is a special kind of information network, which either contains multiple types of objects or multiple types of links.

Definition 1. Heterogeneous information network [9]. A HIN is denoted as $G = \{V, E\}$ consisting of an object set V

and a link set E. A HIN is also associated with an object type mapping function $\phi : V \rightarrow A$ and a link type mapping function $\varphi : E \rightarrow R$. A and R denote the sets of predefined object and link types, where $|A| + |R| > 2$.

Definition 2. Network schema [9]. The network schema is denoted as $S = (A, R)$. It is a meta template for an information network $G = \{V, E\}$ with the object type mapping $\phi : V \rightarrow A$ and the link type mapping $\varphi : E \rightarrow R$, which is a directed graph defined over object types A, with edges as relations from R.

Definition 3. Meta-path [10]. A meta-path ρ is defined on a network schema $S = (A, R)$ and is denoted as a path in the form of $A_1 \overset{R_1}{\rightarrow} A_2 \overset{R_2}{\rightarrow} \cdots \overset{R_l}{\rightarrow} A_{l+1}$ (abbreviated as $A_1 A_2 \ldots A_{l+1}$), which describes a composite relation $R = R_1 \circ R_2 \circ \ldots \circ R_l$ between object A_1 and A_{l+1} , where $\circ$ denotes the composition operator on relations.

A MOOCs Network is a typical heterogeneous network, containing objects from six types of entities/objects: courses ($\mathcal{C}$), teachers ($\mathcal{T}$), users (i.e., students) ($\mathcal{U}$), concepts (i.e., reinforcement learning agents/environments) ($\mathcal{K}$), videos ($\mathcal{V}$) and schools ($\mathcal{S}$). For each course $c \in \mathcal{C}$, it has links to a teacher, a set of users, a set of concepts, a video, and a school. For each teacher $\in \mathcal{T}$, it has links to a set of schools and a set of courses. For each concept $\in \mathcal{K}$, it has links to a set of courses and a set of videos. For each video $\in \mathcal{V}$, it has links to a course, a set of users, and a set of concept. For each user $\in \mathcal{U}$, it has links to a set of courses and a set of videos (the latter is decided by the former). For each school $\in \mathcal{S}$, it has links to a set of teachers and a set of courses (the latter is decided by the former). Note that the link types are defined by these relations.

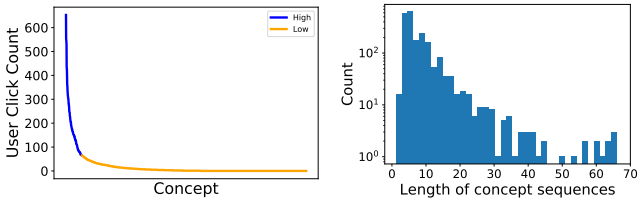
B. Recommender as a MDP

We consider the concept recommendation problem under the standard RL framework. At each time step t, the agent (recommender) observes a state s_t about the environment, and then takes an action (concept) at according to its policy π, which is usually a mapping from states to action probabilities. One time step later, as a result of its action, the agent receives a numerical reward (rating) r_{t+1} and a new state s_{t+1} from the environment. The goal of the agent is to maximize the cumulative reward it receives over T time steps. According to [11], such an RL task can be mathematically described by an MDP, a tuple $(\mathcal{S}; \mathcal{A}; \mathcal{P}; \mathcal{R})$ defined as follows.

$\mathcal{S}$ **is the state space.** The state S_t represents the observed preference of user u at time step t. The representation of state is the n-dimensional user representation getting after the meta-path sampling. $\mathcal{A}$ **is the action space.** We define $\mathcal{A}$ as the set of all concepts, i.e., $\mathcal{A} = \mathcal{I}$. In each state s_t, an action a_t can be taken from the set of available actions $\mathcal{A}(s_t)$, which is defined recursively: $\mathcal{A}(s_t) = \mathcal{A}(s_{t-1}) \backslash \{a_{t-1}\}$ for $t \neq 0$. In other words, the agent is not allowed to choose the concepts that have been recommended at previous time steps. $\mathcal{P}$ **is the transition function.** $\mathcal{P}^a_{ss'} = Pr[s_{t+1} = s' | s_t = s, a_t = a]$ denotes the probability that the environment transits to state s' after receiving action a in state s. In the recommendation setting, the exact transition probabilities are unknown in advance. The agent can observe specific state transitions by interacting with the environment step by step. $\mathcal{R}$ **is the reward function.** $\mathcal{R}^a_{ss'} = \mathbb{E}[r_{t+1} | s_t = s, a_t = a, s_{t+1} = s']$ denotes the expected immediate reward the environment generates after the transition from state s to s' due to action a. In the recommendation setting, the immediate reward of executing an action a only depends on the rating given by user u. Therefore, we define $\mathcal{R}^a_{ss'} = \mathcal{R}_{ua}$.

TABLE I: An overview of *XuetangX* MOOCs dataset.

Nodes	Count	Links	Count
concept	**2,527**	concept-course	21,507
		concept-video	11,732
user	**3,111,637**	user-course	15,045,219
		user-video	53,481,869
course	**7,327**	course-concept	69,012
		course-video	811,841
		course-user	16,724,852
video	**62,191**	video-course	247,433
		video-concept	11,732
		video-user	53,971,707
Total	**3,183,682**	Total	**140,096,904**



(a) Concept click count (b) Concept length distribution

Fig. 2: Data statistical analysis.

C. Problem Formulation

We first declare the notations throughout this paper. Suppose we have U users, K cources V videos and K concepts. Let $\mathcal{U} = \{u_1, u_2, \ldots, u_U\}$ be the set of users, $\mathcal{C} = \{c_1, c_2, \ldots, c_C\}$ be the set of cources, $\mathcal{V} = \{v_1, v_2, \ldots, v_V\}$ be the set of users and $\mathcal{K} = \{k_1, k_2, \ldots, k_K\}$ be the set of concepts. Given a heterogeneous network among users, courses, videos and concepts, as well as the relationship between each two nodes, our goal is to recommend a list of concepts for users to facilitate their online study.

III. Data Analysis

Before introducing our proposed methodology, in this section, we first conduct a comprehensive statistical analysis on a real-world dataset from *XuetangX* MOOCs. We also have a special analysis on the meta-path selection. From the analysis, we introduce some insights which can better motivate our proposed method.

A. Data Analysis

Our proposed approach is evaluated on a real-world dataset obtained from *XuetangX* MOOCs, which currently is one of the largest MOOCs platform in China. This dataset consists of

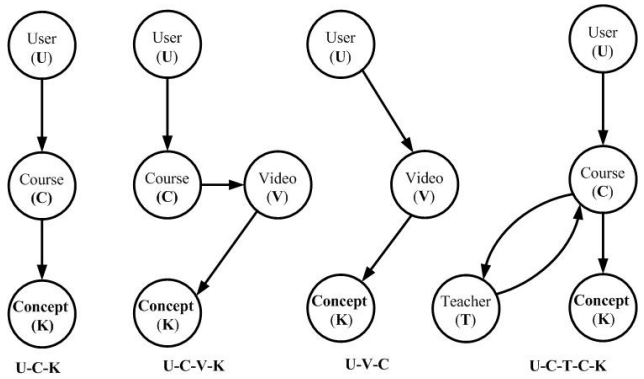


Fig. 3: Analyzed Meta-paths

2,527 concepts, 3,708,461 users, 7,327 courses, 96,950 videos and 140,446,950 relations among them. The detailed statistics of the *XuetangX* MOOCs datasets are shown in Table I.

Fig. 2 shows the statistical analysis of *XuetangX* MOOCs dataset. Fig. 2a shows the distribution of user click count for each concept. We can observe that this distribution follows the long-tail distribution. In other words, the majority of concepts are clicked by few users, which verifies the necessity of concept recommendation. Fig. 2b shows the distribution of the length of concept sequences. We can see that the concepts of most courses are clicked between 10 to 30 times.

B. Meta-path Analysis

Sun et al. [12] point out that meta-path is a powerful tool to systematically define the relation between two object types and capture different semantic meanings of the connection between objects. This paper focuses on RL-based framework for concept recommendation in *XuetangX* MOOCs, and thus, we will not investigate how to automatically learned/obtained meta-paths since meta-paths should be handcrafted by domain experts, as in the literature [13]. According to the literature [14], currently, we analyze and finally obtain four types of meta-path shown in Fig. 3. We describ their semantic meaning as follows:

- **U-C-K** shows that users who learn the same courses will click the similar/related concept labels with target user.
- **U-C-V-K** shows that users who first learn the same concepts and then watch the same videos will click the similar/related concept labels with target user.
- **U-V-K** shows that users who watch the same videos will click the similar/related concept labels with target user.
- **U-C-T-C-K** shows that users who first watch the same courses taught by the same teacher and then learn the same courses will click similar/related concept labels with target user.

IV. Methodology

In this section, we present our proposed methodology in detail, we first give an overview of the whole framework. Then we present our heterogeneous information network for user embedding. Consequently, we introduce our reinforcement learning approach to recommend concepts to users. We also provide some training techniques to train the whole network.

A. An Overview of HIN-RL

Fig. 4 shows an overview of the network architecture of our proposed model. This framework mainly consists of three sub-modules. (a) Meta-path sampling (cf. Sec. IV-B1). In this part, we first construct HIN among users, courses and concepts, and utilize random walk to sample meta-paths on this network for each given article. (b) Meta-path based user embedding with hierarchical attention network (cf. Sec. IV-B2 and Sec. IV-B3). In this part, we propose a hierarchical attention network to embed each math-path into hidden space. On the node-level, we apply a self-attention layer to represent the user with its neighbours. On the path-level, we apply another attention layer to aggregate each path semantic representation to the user. (c) Reinforced concept recommendation. (cf. Sec. IV-C). In this module, we propose a reinforcement learning based approach to recommend courses to users. We will elaborate them in details in the following subsections.

B. Meta-path based User Embedding

1) Meta-path Sampling: To index the network structure of G, we use random walks to sample paths on G. Inspired by DeepWalk [15], we start random walks from each user in our training dataset $\mathcal{D}$ for which concept will be recommended to. For each user in U, we sample N paths. In total, we have $|U| \times N$ paths about the HIN, and we denote the path corpus as $\mathcal{M}$. First, we propose a node-level attention to learn the weight of meta-path based neighbors and aggregate them to get the semantic-specific user embedding. After that, we assign different weights ton meta-paths via path-level attention to get the final representation of the given user.

2) Node-level Attention: For a given meta-path of a specific user, since the node type are heterogeneous and different types of nodes have different feature spaces, we design a projection layer to map them into a common space. In particular, for each type of nodes (e.g., node with type ϕ_i), we design a type-specific transformation matrix M_{ϕ_i} to project the features of different types of nodes into the same feature space. It is worth mentioning that the type-specific transformation matrix is based on node-type rather than edge-type. The projection process can be formularized as follows:

$$\mathbf{h}_\mathbf{i}' = \mathbf{M}_{\phi_i} \cdot \mathbf{h}_i, \tag{1}$$

where $\mathbf{h}_i$ and $\mathbf{h}_i$ are the original and projected feature of node i, respectively. We argue that different node for a given meta-path should contribute different weights to the final user embedding. Therefore, we apply self-attention [16] to learn the weight among various kinds of nodes, which is formularized as follows:

$$\alpha_{ij}^{\Phi} = \frac{\exp\left(\sigma\left(\mathbf{a}_{\Phi}^{\mathrm{T}} \cdot \left[\mathbf{h}_i' \| \mathbf{h}_j'\right]\right)\right)}{\sum_{k \in \mathcal{N}_i^{\Phi}} \exp\left(\sigma\left(\mathbf{a}_{\Phi}^{\mathrm{T}} \cdot \left[\mathbf{h}_i' \| \mathbf{h}_k'\right]\right)\right)}, \tag{2}$$

where σ denotes the activation function, $||$ denotes the concatenate operation and a ϕ is the node-level attention vector for

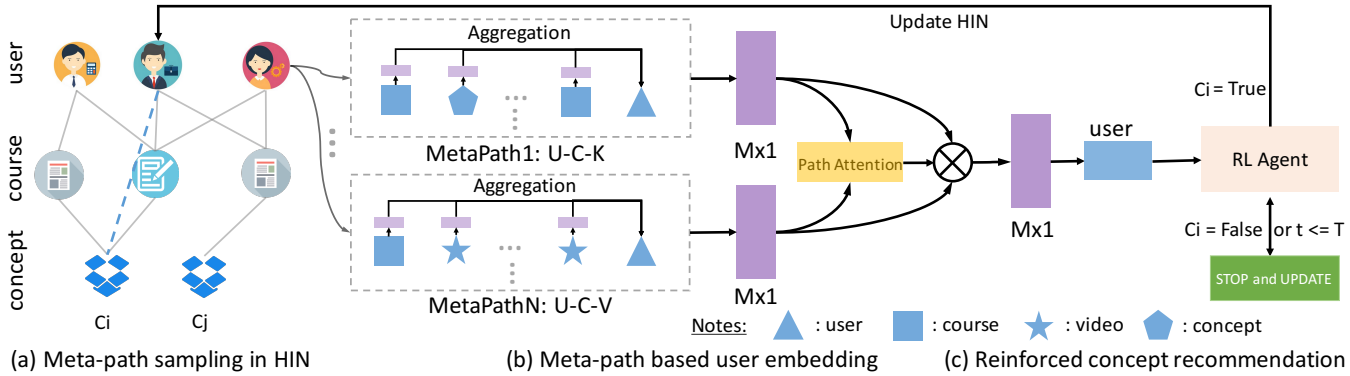


Fig. 4: An overview of our proposed method. (a) Meta-path sampling in Heterogeneous information network (HIN) of MOOCs. (b) Meta-path based user embedding with hierarchical attention networks. (c) Reinforced concept recommendation.

meta-path ϕ. As we can see from Eq. 2, the weight coefficient of (i, j) depends ϕ on their features.

Then, the meta-path based embedding of node i can be aggregated by the neighbor's projected features with the corresponding coefficients as follows:

$$\mathbf{u}_i^{\Phi} = \sigma\left(\sum_{j \in \mathcal{N}_i^{\Phi}} \alpha_{ij}^{\Phi} \cdot \mathbf{h}_j'\right) \tag{3}$$

where z_i is the learned embedding of node i for the meta-path ϕ. Every node embedding is aggregated by its neighbors. Since the attention weight α_{ij}^{ϕ} is generated for single meta-path, it is semantic-specific and able to capture the semantic information.

Since heterogeneous graph present the property of scale free, the variance of graph data is quite high. To tackle the above challenge, we extend node-level attention to multi-head attention so that the training process is more stable [16]. Specifically, we repeat the node-level attention for K times and concatenate the learned embeddings as the semantic-specific embedding:

$$\mathbf{u}_i^{\Phi} = \Big\|_{k=1} \sigma\left(\sum_{j \in \mathcal{N}_i^{\Phi}} \alpha_{ij}^{\Phi} \cdot \mathbf{h}_j'\right) \tag{4}$$

3) Path-leval Attention: To learn the importance of each meta-path, we first transform semantic-specific embedding through a nonlinear transformation (e.g., one-layer MLP). Then we measure the importance of the semantic-specific embedding as the similarity of transformed embedding with a path-level attention vector q. Furthermore, we average the importance of all the semantic-specific node embedding which can be explained as the importance of each meta-path. The importance of each meta-path, denoted as w_{ϕ_i} , is shown as follows:

$$w_{\Phi_i} = \frac{1}{|\mathcal{V}|} \sum_{i \in \mathcal{V}} \mathbf{q}^{\mathrm{T}} \cdot \tanh\left(\mathbf{W} \cdot \mathbf{u}_i^{\Phi} + \mathbf{b}\right) \tag{5}$$

where $\mathbf{W}$ is the weight matrix, $\mathbf{b}$ is the bias vector, $\mathbf{q}$ is the path-level attention vector. Note that for the meaningful comparison, all above parameters are shared for all meta-paths and semantic-specific embedding. After obtaining the importance of each meta-path, we normalize them via softmax function. The weight of meta-path ϕ_i , denoted as β_{ϕ_i} , can be obtained by normalizing the above importance of all meta-paths using softmax function,

$$\beta_{\Phi_i} = \frac{\exp\left(w_{\Phi_i}\right)}{\sum_{i=1}^{P} \exp\left(w_{\Phi_i}\right)} \tag{6}$$

which can be interpreted as the contribution of the meta-path ϕ_i for specific task. Obviously, the higher β_{ϕ_i} , the more important meta-path ϕ_i is. Note that for different tasks, meta-path ϕ_i may has different weights. With the learned weights as coefficients, we can fuse these semantic-specific embeddings to obtain the final embedding $\mathbf{U}$ as follows:

$$\mathbf{U} = \sum_{i=1}^{P} \beta_{\Phi_i} \cdot \mathbf{U}_{\Phi_i} \tag{7}$$

C. Reinforced Concept Recommendation

A common way of building recommendation systems is to estimate a model which minimizes the discrepancy between the model prediction and the immediate user response according to some loss function. In other words, these models do not explicitly take into account the long-term user interest. However, users interest can evolve over time based on what he/she observes, and the recommenders action may significantly influence such evolution. In some sense, the recommender is guiding users interest by displaying particular items and hiding the rest. Moreover, in order to dynamic and personalized recommend the concept, we need the strategy which can change with the environment. Thus, it is more favorable to design a recommendation strategy, such as one based on RL, which can take users long-term interest and dynamic user embedding into account.

We first introduce the simplified setting of concept recommendation as a standard reinforcement learning problem.

The goal of using RL is to find a policy that maximizes the expected cumulative reward:

$$\mathcal{L}_{RL}(\theta) = \mathbb{E}_{\pi_\theta(c|\mathbf{u})} \sum_{t=1}^{T} r_t(c|\mathbf{u}). \tag{8}$$

where r is reward, given the particular user if the predicted concept $\hat{c}$ is true, then the reward will be set as 1, otherwise, it will be -1.

If the recommended concept c_t is correct, our model will add the connection between the user u and the correct recommend concept c_t on HIN. Therefore, we will get the new user embedding u_{t+1} through the previous HIN embedding step. If the recommend concept is correct, HIN-RL will continually recommend new concept, until incorrectly recommending concept or comes to the max timestep T.

However, if the recommend concept is incorrect, the HIN G will not be changed. Thus, the user embedding will not be changed. Since the Q-learning network needs the next step transition u_{t+1} of current step u_t to update, if the recommendation is wrong, then the predict Q_{t+1} will be same with the target Q_t, so Q-learning cannot be used to our setting. Therefore, we seek to use the policy-based RL method policy gradient to update the model.

We can then learn the optimal policy by employing policy gradient. According to the policy gradient theorem [17], [18], the gradient of the expected cumulative reward can be calculated by

$$\nabla_\theta \mathcal{L}_{RL}(\theta) = \sum_{t=1}^{T} [\nabla_\theta \log \pi_\theta(c_t|\mathbf{u}_t)] r_t. \tag{9}$$

We often observed collapse onto a sub-optimal deterministic policy. To avoid the model not being able to explore new concepts that could lead to a better recommendation, we use entropy regularization:

$$\mathbf{H}[\pi_\theta(c|\mathbf{u})] = \sum_{i=1}^{T} \sum_{c_i \in C} \log(\pi_\theta(c_t|\mathbf{u}_t)) \pi_\theta(c_t|\mathbf{u}_t), \tag{10}$$

The final objective is:

$$\mathbb{E}_{c \sim \pi_\theta(c|\mathbf{u})} \mathcal{L}_{RL}(\theta) + \lambda \mathbf{H}[\pi_\theta(c|\mathbf{u})]. \tag{11}$$

where λ is the regularization weight.

Algorithm 1 The Policy Gradient Algorithm (HIN-RL)

Input: training set $\mathcal{U}_{train}$, rating data R, number of episodes K, number of time steps T, discount factor γ, and ϵ-greedy parameter ϵ
Output: the learned recommender policy pi_θ

```
Initialize recommender policy pi_θ with random weights
for episode = 1, ..., K do
    Uniformly pick a user u_0 ∈ U_train as the environment
    Learning the user embedding u by Eq. 7
    set t = 0
    while c_t is correct and t ≤ T do
        Select action a_t using ε-greedy policy w.r.t π_θ
        Take a_t, observe reward r_{t+1}
        Compute user embedding u_{t+1} by Eq. 7
        set t = t+1
    end while
    Update π_θ's weights θ according to Equation 11
end for
```

V. Experiments

A. Experimental Dataset

The dataset is introduced in Section III. We select the enrolled behaviors from October 1st, 2016 to December 30th, 2017 as the training set, and those from January 1st, 2018 to March 31st, 2018 as the test set. Each instance in the training or the test set is a sequence of historical enrolled courses paired with a target course. During the training process, for each sequence in the training data, we hold out the last course as the target course, and the rest are treated as the historical courses. For each positive instance, we construct 4 negative instances by replacing the target course with each of 4 randomly sampled courses. During the test process, we treat each concept which is related to enrolled courses in the test set as the target concept, and the corresponding concepts of the same user in the training set as the historical concepts. Also, a user often continuously clicks the same concept label and then generates multiple clicked concept records. These records are treated as a single record. Each positive instance in the test set is paired with 99 randomly sampled negative instances [19].

B. Evaluation Metrics

For our evaluation, several categories of Web search evaluation metrics are used to measure the performance of our proposed model from different aspects, including some relevance based metrics and ranking based metrics. To measure the relevance of our recommended results, we use the hit ratio at rank k ($HR@k$), which is widely used and is defined as $HR@K = \frac{\#Hits@K}{|GT|}$, where $|GT|$ is the size of testing dataset. $HR@k$ measures the percentage of successes in prediction to the number of attempts. Another ranking metric we use in our evaluation is normalized discounted cumulative gain (NDCG), which measures the performance of a retrieval system based on the graded relevance of the retrieved entities. The NDCG@k is defined as $NDCG@k = \frac{1}{|Q|} \sum_{q=1}^{|Q|} Z_{kq} \sum_{j=1}^{k} \frac{2^{r(j)}-1}{log(1+j)}$, where Z_{kq} is a normalization factor so that a perfect ranking's $NDCG$ at k for query q is 1; $r(j)$ is the relevance score assessors given to a retrieved entity for query q. We use Mean Reciprocal Rank (MRR) to evaluate the ranking of our search results. A larger MRR value means a better result. The MRR is defined as $MRR = \frac{1}{|Q|} \sum_{i=1}^{|Q|} \frac{1}{rank_i}$, where $|Q|$ is the size of query set. In addition, another metric is the area under the ROC curve (AUC), which measures the quality of recommendation ranking.

C. Baselines

We compare our proposed model with the following state-of-the-art baselines.

- **BPR** [20] optimizes a pairwise ranking loss for the recommendation task in a Bayesian way.

- **MLP** [21] applies a multi-layer perceptron (MLP) on a pair of user and concept embeddings to learn the probability of recommending the concept to the user.
- **FM** [22] is a principled approach that can easily incorporate any heuristic features. For fair comparison, we only use the embeddings of users and concepts.
- **FISM** [23] is an item-to-item collaborative filtering algorithm which conducts recommendation based on the average embedding of all the historical concepts and the embedding of the target concept.
- **NAIS** [19] is also an item-to-item collaborative filtering algorithm but distinguishes the weights of different historical concepts by an attention mechanism.
- **NASR** [24] is an improved GRU model that estimates an attention coefficient for each historical concept based on the corresponding hidden vector output by GRU.
- **HIN-SL** represents our proposed model which only applies a heterogeneous information networks for user embedding, without reinforcement learning.
- HIN-RL represents our proposed model which combines the heterogeneous information networks for user embedding and the reinforcement learning for better recommendation.

D. Implementation Details

We train all our model on a Nvidia GeForce GTX1080Ti GPU card with 11GB RAM. We divide the datasets twice for evaluating our model: 1) 80% training set and 20% test set, and 2) 60% training set and 40% test set. The dimension size of final user embedding is set to 64. Reinforcement learning part of our proposed algorithms is consisted of a hidden layer with 128 units. We first pre-train the model using the cross-entropy loss function with 10,000 episodes. The mini-batch size is set to 8 while the learning rate is set to 0.001. After that, the loss function is adjusted to be Policy Gradient (PG) policy and at this time, the learning rate is changed to 0.0001. To avoid over-fitting, we employ regularization rule and set λ of regularization rate to 0.08. The sequence length of recommended concepts automatically keep consistent with the mini-batch size. Through the entire training of our HIN-RL model, we update the parameter using ADAM optimizer.

E. Experimental Results

Table II shows the experimental results of comparison between our proposed model and some state-of-the-art approaches on the offline datasets. From this table, we can find that our proposed model outperforms other baselines in all evaluation metrics. When comparing the HIN-SL/RL with some baselines, we can see that our proposed heterogeneous information network really contains more information of users. When comparing HIN-SL with HIN-RL, we can see that the proposed reinforcement learning also has a positive effect on the performance based on the supervised learning. This can be illustrated by the fact that the exploration mechanism in reinforcement learning can lead our model to a better optimized result.

F. Parameters Analysis

Here, we investigate the sensitivity of parameters and report the results of HIN-RL with various parameters in Fig. 5 and Fig. 6.

1) Attention Head K: In order to verify the effectiveness of multi-head attention involved in our model, we analyze the performance of the HIN-RL model when the number of attention head varies. The number of attention head are set to 2, 4, 6, 8, 10 and 12 in sequence. The experimental results are shown in Fig. 5. We can see that the growth of the number slightly improves the performance of our model. Although the attention mechanism included in our model consumes a little more the training time, the results prove that it can make our model both more reliable and accurate.

2) Regularization Rate λ: We conduct many experiments about entropy regularization to validate the effectiveness of the parameter λ. We first set λ to 0.01, 0.1, 1 and 10 and give their experimental results in the four bigger figures of Fig. 6. And then, for more fine-grained analyses, we set λ from 0.1 to 1.0 (i.e., 0.02, 0.04, 0.06 and 0.08) and present the results in the four smaller figures each of which is contained in the corresponding bigger one, respectively. From Fig. 6, we can find that λ's value over 0.01 and 0.1 can help our HIN-RL model achieve better performance of concepts recommendation, otherwise the model will gain the lower performance.

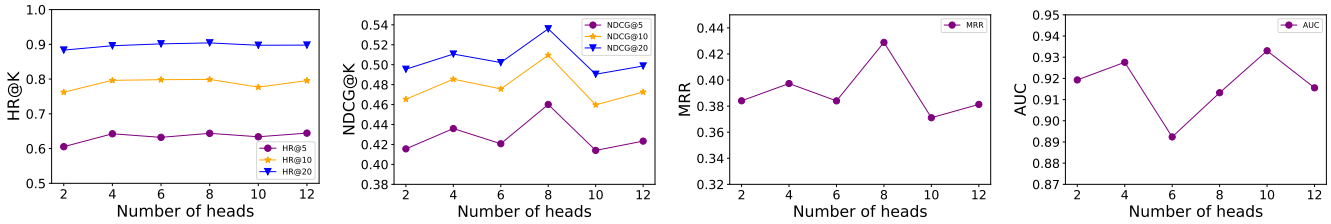
G. Case Study

We conduct case study to demonstrate the effectiveness of our proposed method. In this part, we aim to provide insights pertaining to the results of our model. One of the advantages is that our HIN-RL model learns the individual user's study pattern through user embedding with meta-paths. We use the model to predict the next concept with users' different clicked history and the current same clicked item.

We randomly select user A (ID: 6806276) and user B (ID: 8136117) to feed our model of HIN-RL. The experiments take 182ms to finish and involve 937 related concept labels. Fig. 7 illustrates the recommended concepts with diversity. An interesting result can be found through the Figure that users with different backgrounds (or learning experiences) get a diverse and personalized list of concepts recommendation even if they click/learn the same concept label (i.e., *Operating System*). For example, in *Xuetang* MOOCs, the interest/preference of user A is about algorithm-related course, thus the concepts of *Sequence*, *Divide and Conquer*, *Towers of Hanoi*, etc. are recommended to him/her, and finally the user click/select *Towers of Hanoi* concept. What this case illustrates keeps consistent with the results given by our model. The case study clearly shows that our model of HIN-RL can dynamically recommend different concepts to users according to their different browsing history.

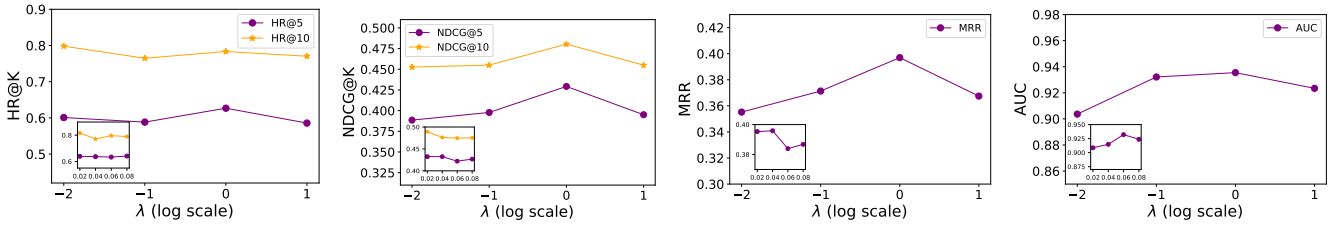
TABLE II: Quantitative results (%) of experiments on *XuetangX* MOOCs datasets. (Best scores are in boldface.)

Training	Model	HR@5	HR@10	HR@20	NDCG@5	NDCG@10	NDCG@20	MRR	AUC
60%	**BPR**	41.50	56.28	72.12	28.95	33.74	37.76	28.68	84.34
	MLP	32.75	52.66	69.88	17.67	25.32	30.71	18.34	82.73
	FM	36.03	54.40	72.39	23.99	29.89	34.44	24.46	83.20
	FISM	36.41	54.31	70.79	20.15	27.73	33.83	21.65	84.24
	NAIS	36.47	57.92	77.99	21.93	28.10	34.12	22.15	86.76
	NASR	38.67	53.21	67.13	21.19	27.84	32.93	19.687	79.90
	HIN-SL1	59.86	75.61	88.55	40.54	45.69	48.99	37.57	90.98
	HIN-SL2	65.30	80.42	91.68	45.88	50.8	53.68	42.54	**93.79**
	HIN-RL1	63.44	75.33	89.86	43.60	47.39	51.12	39.91	90.75
	HIN-RL2	**74.68**	**84.55**	**96.13**	**55.44**	**58.64**	**61.55**	**51.23**	93.56
80%	**BPR**	46.33	62.46	79.66	32.17	37.36	41.51	31.56	86.10
	MLP	36.80	58.99	72.37	22.31	29.26	34.41	21.46	85.95
	FM	40.57	58.67	76.44	36.55	39.68	39.30	30.67	85.74
	FISM	58.49	76.10	74.89	37.60	42.03	42.79	32.93	85.32
	NAIS	41.12	66.24	86.49	23.82	32.01	37.93	23.92	88.63
	NASR	44.37	62.15	74.75	23.64	31.72	38.21	21.17	82.15
	HIN-SL1	60.92	77.91	90.79	40.54	46.07	49.34	37.31	92.35
	HIN-SL2	64.14	77.7	90.15	41.73	47.03	50.25	38.63	92.01
	HIN-RL1	64.13	78.92	89.55	42.68	47.54	48.86	38.10	91.15
	HIN-RL2	**73.73**	**89.83**	**96.61**	**52.3**	**57.59**	**59.31**	**47.93**	**95.76**



(a) Impact of K on $HR@K$ (b) Impact of K on $NDCG@K$ (c) Impact of K on MRR (d) Impact of K on AUC

Fig. 5: Parameter sensitivity of HIN-RL -Attention Head K.



(a) Impact of λ on $HR@K$ (b) Impact of λ on $NDCG@K$ (c) Impact of λ on MRR (d) Impact of λ on AUC

Fig. 6: Parameter sensitivity of HIN-RL - λ.

VI. Related Work

A. Mining in MOOCs

A large number of big-data-mining issues or tasks had been effectively solved by employing Heterogeneous Information Network (HIN) such as recommendation system [25], activity prediction [26], community detection [27] and etc. Currently, DBLP[1] (a bibliographic information network) and IMDB[2] (Online Movie DataBases) are commonly used as typical information network datasets to help researchers demonstrate their proposed algorithms or methods. However, as a newly emerging HIN, MOOCs are more typical and contain richer semantics in objects and links, and thus they form a new development of data mining. Among previous works on MOOCs, such as course concept extraction [2], learning behavior predicting [3], course recommendation [4] [5], et al, the course recommendation is the most similar work to us which employs a hierarchical reinforcement learning algorithm to revise the user profiles and tune the course recommendation model on the revised profiles. However, our work is different from it in many aspects. Our work considers MOOCs as a large HIN with meta-path-based semantic information to recommend more fine-grained knowledge to users. Furthermore, we integrate attention-based embedding of heterogeneous information network with an extended deep reinforcement learning framework to recommend knowledge concepts.

[1]http://www.informatik.uni-trier.de/ ley/db/

[2]https://datasets.imdbws.com/

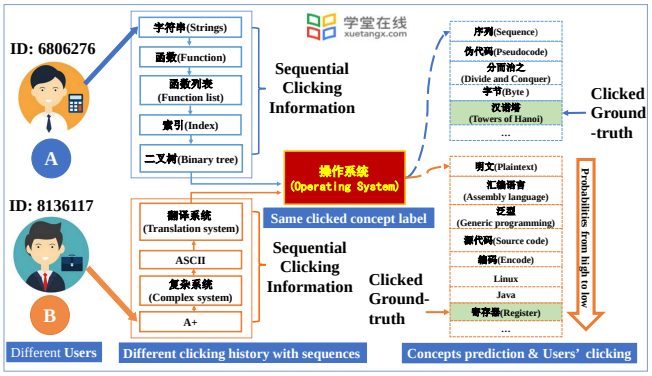


Fig. 7: Diversity recommendation case with sequential clicked concepts information in a dynamic learning environment of *XuetangX* MOOCs.

B. HIN-based Recommendations

E-commerce applications or online systems/platforms (e.g., Amazon,eBay, Taobao[3] and *XuetangX* MOOCs [4]) include a lot of object types (e.g., all kinds of commodities, users, courses, teachers, videos, schools, and knowledge concepts) and the rich relations among object types, which naturally constitute a HIN. The comprehensive information integration and rich semantic information of HIN make it promising to generate better recommendation. Sun et al. [10] first propose to explore the meta-structure of the information network, i.e., put forward to the concepts of meta-paths to systematically capture numerous semantic relationships across multiple types of objects, which are defined as a path over the graph of network schema. Meta-paths can provide guidance for search and mining of the network (e.g., recommendation) and help analyze and understand the semantic meaning of the objects and relations in the network [12]. HIN-based methods have shown superior performance than other existing methods due to its excellent ability in modeling heterogeneous information. Cai et al. [28] proposed GAN-HBNR based on Generative Adversarial bibliographic Network module for citation recommendation. Cen et al. [29] proposed an embedding method for the attributed multiplex heterogeneous Network to improve the performance of Alibaba recommendation system. Recently, HAN [30], utilizes attention mechanism in entity and meta-path respectively, has shown its effectiveness in HIN embedding. In particular, the attention mechanism is widely exploited in the representation of HIN. For example, HERec [31] and MCRec [32] integrate attention mechanism and HIN to learn network information representations for recommendation tasks. Attention-based models have shown their potential in the recommendation of HIN. Different from the proposed models for citation, movie and product recommendation, our method focuses on a complex heterogeneous network in *XutangX* MOOCs for knowledge concept recommendation. In this paper, we combine an extended deep reinforcement learning framework with the representations of users status to implement a recommendation of user interested knowledge concepts.

[3]http://www.taobao.com/

[4]http://www.xuetangx.com/

C. Reinforcement Learning on Recommender System

Reinforcement learning has been widely used in recommendation field [8] [33] [34] [7]. There are several RL works on movies recommendation [35], news recommendation [34], music recommendation [36], etc. However, these methods have two major issues. First, they fail to capture semantic relations among objects, but usually employ the rating scores between users and items in recommender system. Thus, they recommend the similar objects, which may cause users to get bored. Second, few studies consider using an attention mechanism in order to model the case when a user has interests in many different courses. Different from existing RL-based recommendation work [37] [38], our model leveraging rich meta-path based context, which is able to learn interaction-specific representations for students, courses and concept items etc. To our knowledge, it is the first time that meta-path based context, i.e., <student, meta-path, concept>, has been explicitly modeled in a RL-based interaction model, for the task of multi-round concepts recommendation in MOOCs.

VII. Conclusions and Future Work

We present HIN-RL, the first attempt to solve the problem of personalized (knowledge) concepts recommendation in MOOCs platform. We define the meta-path schema to model the relation between two object types and capture different semantic meanings of the connection between objects. We embed *XuetangX* MOOCs into the low-dimensional feature vector through adopting a meta-path guided random walk strategy to generate path instances. We present a graph attention based on a RNN model equipped with an external memory component to improve the aforementioned embedding presentation and focus on small but informative parts of *XuetangX* MOOCs. In order to improve the dynamic nature of news characteristics and user preference, we propose to use reinforcement method which can consider current reward and future reward simultaneously. Experiments have shown that our method can improve the recommendation accuracy and recommendation diversity significantly. Our method can be generalized to many other recommendation problems.

In the near future, we will deploy our concept recommendation algorithms on real-world MOOCs platform *XuetangX* MOOCs and provide a commercial concepts recommendation service with personalized and dynamic characteristics. We will also explore how to use Inverse Reinforcement Learning (IRL) to describe the system and further improve the performance of our method.

References

[1] L. Pan, C. Li, J. Li, and J. Tang, "Prerequisite relation learning for concepts in moocs," in *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, 2017, pp. 1447–1456.

[2] L. Pan, X. Wang, C. Li, J. Li, and J. Tang, "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)*, 2017, pp. 875–884.

[3] J. Qiu, J. Tang, T. X. Liu, J. Gong, C. Zhang, Q. Zhang, and Y. Xue, "Modeling and predicting learning behavior in moocs," in *Proceedings of the ninth ACM international conference on web search and data mining*. ACM, 2016, pp. 93–102.

[4] X. Jing and J. Tang, "Guess you like: course recommendation in moocs," in *Proceedings of the International Conference on Web Intelligence*. ACM, 2017, pp. 783–789.

[5] J. Zhang, B. Hao, B. Chen, C. Li, H. Chen, and J. Sund, "Hierarchical reinforcement learning for course recommendation in moocs," *Psychology*, vol. 5, no. 4.64, pp. 5–65, 2019.

[6] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua, "Neural collaborative filtering," in *Proceedings of the 26th International Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 2017, pp. 173–182.

[7] Y. Lei and W. Li, "When collaborative filtering meets reinforcement learning," *arXiv preprint arXiv:1902.00715*, 2019.

[8] X. Zhao, L. Zhang, Z. Ding, D. Yin, Y. Zhao, and J. Tang, "Deep reinforcement learning for list-wise recommendations," *arXiv preprint arXiv:1801.00209*, 2017.

[9] Y. Sun and J. Han, "Mining heterogeneous information networks: principles and methodologies," *Synthesis Lectures on Data Mining and Knowledge Discovery*, vol. 3, no. 2, pp. 1–159, 2012.

[10] Y. Sun, J. Han, X. Yan, P. S. Yu, and T. Wu, "Pathsim: Meta path-based top-k similarity search in heterogeneous information networks," *Proceedings of the VLDB Endowment*, vol. 4, no. 11, pp. 992–1003, 2011.

[11] R. S. Sutton and A. G. Barto, *Reinforcement learning: An introduction*. MIT press, 2018.

[12] Y. Sun and J. Han, "Meta-path-based search and mining in heterogeneous information networks," *TSINGHUA SCIENCE AND TECHNOLOGY*, vol. 18, no. 4, pp. 329–338, 2013.

[13] C. Luo, W. Pang, Z. Wang, and C. Lin, "Hete-cf: Social-based collaborative filtering recommendation using heterogeneous relations," in *2014 IEEE International Conference on Data Mining*, Dec 2014, pp. 917–922.

[14] Y. Sun, B. Norick, J. Han, X. Yan, P. S. Yu, and X. Yu, "Pathselclus: Integrating meta-path selection with user-guided object clustering in heterogeneous information networks," *ACM Trans. Knowl. Discov. Data*, vol. 7, no. 3, pp. 11:1–11:23, Sep. 2013. [Online]. Available: http://doi.acm.org/10.1145/2500492

[15] B. Perozzi, R. Al-Rfou, and S. Skiena, "Deepwalk: Online learning of social representations," in *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*. ACM, 2014, pp. 701–710.

[16] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," in *Advances in neural information processing systems*, 2017, pp. 5998–6008.

[17] R. J. Williams, "Simple statistical gradient-following algorithms for connectionist reinforcement learning," *Machine learning*, vol. 8, no. 3-4, pp. 229–256, 1992.

[18] R. S. Sutton, D. A. McAllester, S. P. Singh, and Y. Mansour, "Policy gradient methods for reinforcement learning with function approximation," in *Advances in neural information processing systems*, 2000, pp. 1057–1063.

[19] X. He, Z. He, J. Song, Z. Liu, Y. Jiang, and T. Chua, "NAIS: neural attentive item similarity model for recommendation," *IEEE Trans. Knowl. Data Eng.*, vol. 30, no. 12, pp. 2354–2366, 2018.

[20] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme, "BPR: bayesian personalized ranking from implicit feedback," *CoRR*, vol. abs/1205.2618, 2012.

[21] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T. Chua, "Neural collaborative filtering," *CoRR*, vol. abs/1708.05031, 2017.

[22] S. Rendle, "Factorization machines with libfm," *ACM TIST*, vol. 3, no. 3, pp. 57:1–57:22, 2012.

[23] S. Kabbur, X. Ning, and G. Karypis, "FISM: factored item similarity models for top-n recommender systems," in *The 19th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD 2013, Chicago, IL, USA, August 11-14, 2013*, 2013, pp. 659–667.

[24] J. Li, P. Ren, Z. Chen, Z. Ren, T. Lian, and J. Ma, "Neural attentive session-based recommendation," in *Proceedings of the 2017 ACM on Conference on Information and Knowledge Management, CIKM 2017, Singapore, November 06 - 10, 2017*, 2017, pp. 1419–1428.

[25] B. Hu, C. Shi, W. X. Zhao, and T. Yang, "Local and global information fusion for top-n recommendation in heterogeneous information network," in *Proceedings of the 27th ACM International Conference on Information and Knowledge Management*. ACM, 2018, pp. 1683–1686.

[26] J. Ye, Z. Zhu, and H. Cheng, *What's Your Next Move: User Activity Prediction in Location-based Social Networks*, 05 2013, pp. 171–179.

[27] C. Tu, X. Zeng, H. Wang, Z. Zhang, Z. Liu, M. Sun, B. Zhang, and L. Lin, "A unified framework for community detection and network representation learning," *IEEE Trans. Knowl. Data Eng.*, vol. 31, no. 6, pp. 1051–1065, 2019.

[28] X. Cai, J. Han, and L. Yang, "Generative adversarial network based heterogeneous bibliographic network representation for personalized citation recommendation," in *Thirty-Second AAAI Conference on Artificial Intelligence*, 2018.

[29] Y. Cen, X. Zou, J. Zhang, H. Yang, J. Zhou, and J. Tang, "Representation learning for attributed multiplex heterogeneous network," *arXiv preprint arXiv:1905.01669*, 2019.

[30] X. Wang, H. Ji, C. Shi, B. Wang, P. Cui, P. Yu, and Y. Ye, "Heterogeneous graph attention network," *arXiv preprint arXiv:1903.07293*, 2019.

[31] C. Shi, B. Hu, W. X. Zhao, and P. S. Yu, "Heterogeneous information network embedding for recommendation," *IEEE Transactions on Knowledge and Data Engineering*, vol. 31, no. 2, pp. 357–370, 2019.

[32] B. Hu, C. Shi, W. X. Zhao, and P. S. Yu, "Leveraging meta-path based context for top-n recommendation with a neural co-attention model," in *Proceedings of the 24th ACM SIGKDD International Conference on Knowledge Discovery & Data Mining*. ACM, 2018, pp. 1531–1540.

[33] X. Zhao, L. Xia, L. Zhang, Z. Ding, D. Yin, and J. Tang, "Deep reinforcement learning for page-wise recommendations," in *Proceedings of the 12th ACM Conference on Recommender Systems*. ACM, 2018, pp. 95–103.

[34] G. Zheng, F. Zhang, Z. Zheng, Y. Xiang, N. J. Yuan, X. Xie, and Z. Li, "Drn: A deep reinforcement learning framework for news recommendation," in *Proceedings of the 2018 World Wide Web Conference on World Wide Web*. International World Wide Web Conferences Steering Committee, 2018, pp. 167–176.

[35] F. Liu, R. Tang, X. Li, W. Zhang, Y. Ye, H. Chen, H. Guo, and Y. Zhang, "Deep reinforcement learning based recommendation with explicit user-item interactions modeling," *arXiv preprint arXiv:1810.12027*, 2018.

[36] X. Wang, Y. Wang, D. Hsu, and Y. Wang, "Exploration in interactive personalized music recommendation: a reinforcement learning approach," *ACM Transactions on Multimedia Computing, Communications, and Applications (TOMM)*, vol. 11, no. 1, p. 7, 2014.

[37] F. Pan, Q. Cai, P. Tang, F. Zhuang, and Q. He, "Policy gradients for contextual recommendations," 2019.

[38] X. Chen, S. Li, H. Li, S. Jiang, Y. Qi, and L. Song, "Neural model-based reinforcement learning for recommendation," *arXiv preprint arXiv:1812.10613*, 2018.