IEEE Access
Multidisciplinary : Rapid Review : Open Access Journal

Received 4 November 2022, accepted 15 December 2022, date of publication 28 December 2022, date of current version 1 February 2023.

Digital Object Identifier 10.1109/ACCESS.2022.3232552

RESEARCH ARTICLE

ROME: A Graph Contrastive Multi-View Framework From Hyperbolic Angular Space for MOOCs Recommendation

HAO LUO[1,2], NOR AZURA HUSIN[1], AND TEH NORANIS MOHD ARIS[1]
[1]Faculty of Science Computer and Information Technology, Universiti Putra Malaysia, Serdang, Selangor 43400, Malaysia
[2]School of Big Data, University of Baoshan, Baoshan 10686, China

Corresponding author: Nor Azura Husin (n_azura@upm.edu.my)

This work was supported in part by the Baoshan University School Transformation Development Project under Grant ZKZX202001, and in part by the Ministry of Higher Education under Grant FRGS FRGS/1/2020/ICT02/UPM/02/2(08-01-20-2315FR).

ABSTRACT As Massive Open Online Courses (MOOCs) expand and diversify, more and more researchers study recommender systems that take advantage of interaction data to keep students interested and boost their performance. In a typical roadmap, courses and videos are recommended using a graph model, but this does not take into account the user's learning needs with some particular subjects. However, all existing graph models degrade performances either by ignoring the data sparsity issue caused by a large number of concepts, which may lead to biased recommendations, or by constructing improper contrasting pairs, which may result in graph noise. To overcome both challenges, we propose a gRaph cOntrastive Multi-view framEwork (ROME) from hyperbolic angular space to learn user and concept representations based on user-user and concept-concept relationships. The first step is to use hyperbolic and Euclidean space representations as different views of graph and maximize the mutual information between them. Furthermore, we maximize the angular decision margin in graph contrastive training objects to enhance pairwise discriminative power. Our experiments on a large-scale real-world MOOC dataset show that the proposed approach outperforms several baselines and state-of-the-art methods for predicting and recommending concepts of interest to users.

INDEX TERMS Recommender system, graph contrastive learning, manifold learning.

I. INTRODUCTION

In recent years, deep neural networks have been widely applied to recommendation systems, especially the graph neural networks (GNNs) in recent [1], [2], [3], [4]. As for online education, recommendation systems based on GNNs that recommend courses or videos to users have also been studied and deployed to keep users' interest in MOOCs. The overall graph relationship is shown in **Figure 1**. Nevertheless, recommending courses and videos on a wide range of knowledge concepts ignores user interests and the learning needs associated with some specific knowledge concepts. Therefore, increasing attention has been paid to recommending knowledge concepts of interest to users [5], [6].

The associate editor coordinating the review of this manuscript and approving it for publication was Fabrizio Messina.

Traditional graph recommender systems only consider the simple user-concept interaction data to predict or recommend the concepts of the user interest. Typically, a collaborative filtering system [7], [8] learns user interest and estimates preferences from collected user behavior data, which is a popular framework for building recommender systems. Despite their intuitive nature and excellent explanation, these methods cannot effectively deal with graph relationship, resulting in a unpromising effect. Another famous road-map is to utilize the matrix factorization, which can relieve the sparsity to some extent. This kind of methods obtains the implicit vectors of users and courses on the basis of co-occurrence matrix [8], [9], [10]. A predicted score and a list of recommended courses are calculated from the implicit

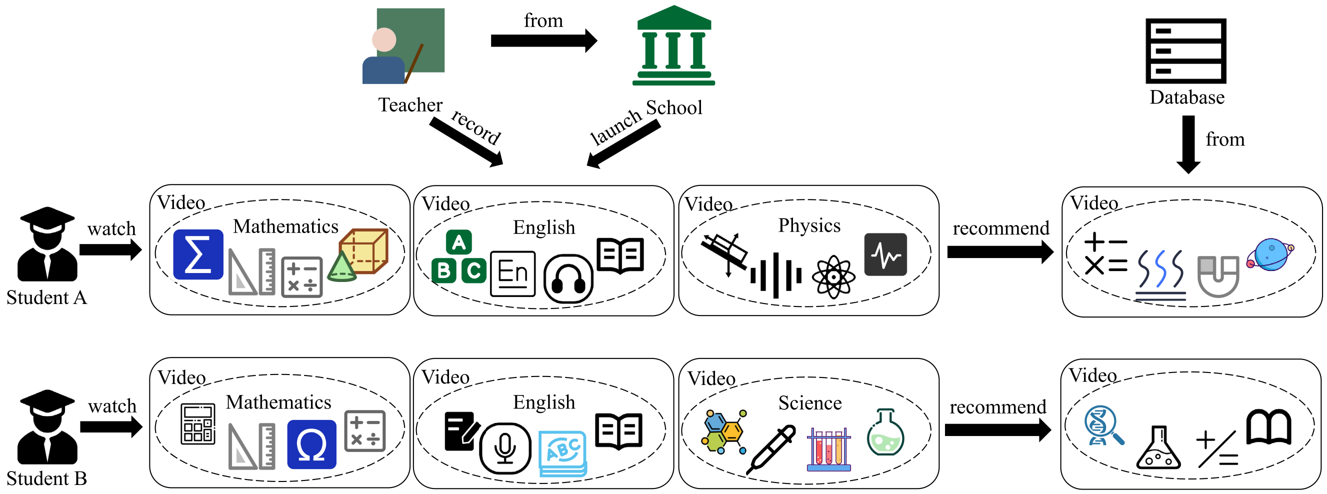

FIGURE 1. **An example of the heterogeneity in MOOCs. MOOCs have different types of entities and relationships between them. Each dashed circle portrays an identity of a course with many concepts. And each rectangle represents an online video. In the case of student A, the concept is of interest if the student has previously learned it or will learn it in the near future.**

vectors when recommending courses to users. Users, courses, and contexts cannot be easily added, so additional valid information can't be changed.

Recently, GNNs have been proposed to solve the drawbacks of the traditional recommender systems and the performance of the recommendation system is greatly improved. Although these methods usually achieve state-of-the-art recommendation performance, there still exist some deficiencies that can be improved. Firstly, the sparsity problem of user-concept interactions. As it is shown in [5], it assumes that the relationship on MOOCs platform corresponds to a homogeneous graph that can be used to learn the representations of the users and the concepts with respect to the chosen path, as a result, traditional GNNs can be implemented on the graph to learn the representations/embeddings of users and concepts. As there are a high number of resources and increasing variety of data, as well as an equally high number of knowledge concepts in existence, the data sparsity problem degrades the performance inevitably.

Secondly, over-fitting on the noise injected into the graph view. In the vanilla recommender system community, the recent state-of-the-art attempts have been made to use contrastive learning in order to solve the sparsity problem related to the user-item data [11]. It optimizes the contrastive loss InfoNCE [12] to learn more uniform user-item representations, which discards the graph augmentations and instead adds uniform noises to the embedding space for creating contrastive views. This operation indeed avoids the challenges on the construction of proper contrasting pair. However, noise injection inevitably causes over-fitting on the graph topology relationship, which interferes with the learning performance of the recommender system. Different from the existing recommender system methods, we abandon the noisy contrastive pair construction and utilize the representation from dual spaces as the contrastive expressiveness to boost the performance.

In this paper, to remedy above issues, we propose a gRaph cOntrastive Multi-view framEwork (ROME) from hyperbolic angular space to learn user and concept representations based on user-user and concept-concept relationships. In order to address both the sparsity problem of user-concept interactions as well as the noise generation issue, we propose to implement the graph representation on hyperbolic space, based on the recent progression of geometric graph mining in hyperbolic spaces [13], [14]. And our graph contrastive targets maps the representations to the angular space for more discriminative ability. As a multi-view framework, using a hyperbolic view and an Euclidean view as embedding spaces, we generate multiple views of the input graph by taking advantage of their greater expressiveness. The hyperbolic space has two major advantages. First, hyperbolic embeddings have greater expressiveness than their Euclidean counterparts, since hyperbolic spaces require less space to accommodate graphs with complex structures. Hyperbolic space expands exponentially in contrast to Euclidean space, which expands polynomially. This will lead to purer, compact, but powerful embedding spaces if we combine it with much lower dimensional embeddings. Second, hyperbolic space is more suitable for capturing hierarchical structures in graph data, which makes hyperbolic space suitable for graph contrastive learning paradigms as a special view.

Furthermore, we propose an approach to enhance pairwise discrimination power by maximising the decision margin for the contrastive learning target in the angular space. It is well known that most of the targets focus only on constructing positive and negative representation pairs and are not concerned with the training objective, such as NT-Xent [15], [16], which is not sufficient for achieving discrimination and cannot handle partial semantic ordering.

According to our experiments using the real-world MOOCs dataset, our proposed approach outperforms several baselines and existing methods for predicting and recommending concepts of interest to users. The main contribution of this paper is summarized as follows:

- We propose a novel graph contrastive multi-view Framework from hyperbolic angular space named ROME for MOOCs recommendation.
- Our ROME firstly uses hyperbolic and Euclidean space representations as different views of graph and maximize the mutual information between them. Furthermore, we maximize the angular decision margin in graph contrastive training objects to enhance pairwise discriminative power.
- Extensive experiments on the large-scale real-world MOOC datasets fully validate the effectiveness of our ROME beyond existing baselines for predicting and recommending concepts of interest to users.

The remaining part of this study is structured as follows: Section II introduces the related work on the graph contrastive learning and the recommender system. Section III covers the background knowledge of this study. Section IV is the key part of our research method ROME with its details. Section V shows the performance evaluation of the proposed ROME, including the experimental setting, data description, model comparison and discussion. Section VI summarizes the experimental results and provides appropriate suggestions for future research content.

II. RELATED WORK

A. GRAPH CONTRASTIVE LEARNING

As for contrastive learning in graph-related tasks, it works in a self-supervised manner [17] and it is inherently a possible solution to the data sparsity issue in recommender systems, especially with the development of GNNs [18]. DGI's [19] and InfoGraph's [20] are among the early works that apply the concept of local-global contrastive objectives [21] to node and graph representation learning by contrasting pairs of node/graph elements. After that, MVGRL [22] is used to generate views of the original graph by utilizing fixed diffusion operations such as heat kernels [23] and Personal PageRank [24]. As a result of adopting the local-global contrastive objective, the MVGRL achieves state-of-the-art performance both on the node classification task as well as on the graph classification task. Inspired by MoCo, GCC [25] generates node views through sub-graph sampling with random walks, where the different sub-graphs are taken as negatives. However, all of the methods above are confused by constructing high-quality contrastive pairs in graph.

B. GRAPH-BASED RECOMMENDATION SYSTEM

Generally, recommender systems are designed to predict whether a user will interact with an item, for example, by clicking on it, rating it, or purchasing it. As such, collaborative filtering (CF), which focuses on exploiting the past user-item interactions to achieve the prediction, remains to be a fundamental task towards effective personalized recommendation [26], [27], [28]. With the development of GNNs, the powerful representation ability of graph topology boosts the development of recommender. Our work is inspired by several recent efforts that provide deep insights into GNNs. Specifically, [29] suggests that GCN is overcomplicated and presents the SGCN model, which eliminates non-linearities and collapses multiple weight matrices into one. There is a significant difference between LightGCN [30] and SGCN, the main reason being that they are developed for different tasks, and the rationale for simplifying models is therefore different. LightGCN, on the other hand, is based on collaborative filtering (CF), where each node only has an ID feature. In terms of node classification accuracy, SGCN is on par with (sometimes weaker than) GCN. When it comes to the accuracy of CF, LightGCN outperforms GCN by a considerable margin (an improvement of over 15 percent over NGCF). Thus, how to leverage the graph representations to boost the recommender system attracts more and more attention in this community.

III. PRELIMINARY

A. DEFINITION OF META-PATH

Using a set of learned concepts, course information, videos, and other contextual information, we predict and recommend concepts that a student might be interested in based on their watching history. Given n users $\mathcal{U} = \{u_1, \ldots, u_n\}$ and m concepts $\mathcal{C} = \{c_1, \ldots, c_m\}$, we define an implicit feedback matrix $R \in \mathbb{R}^{n\times m}$ with each entry $r_{u,c} = 1$ if u has learned c and $r_{u,c} = 0$ otherwise following [5]. In order to approach the problem, it may be useful to frame it in terms of the heterogeneous graph, which is denoted as $\mathcal{G} = \{\mathcal{V}, \mathcal{E}\}$ including a object set $\mathcal{V}$ and a link set $\mathcal{E}$. Furthermore, the heterogeneous graph is associated with a mapping function for the object type $\phi : \mathcal{V} \rightarrow \mathcal{O}$ and a mapping function for the link type $\psi : \mathcal{E} \rightarrow \mathcal{R}$. According to [31], R indicates predefined types of objects and links, where $|\mathcal{O}| + |\mathcal{R}| > 2$.

We can represent the MOOC database in our study as a heterogeneous graph. User, concept, video, course, school, and teacher are among the six common types of entities in the heterogeneous graph. Those entities are also linked together through links that describe their relationships to form the graph. A network schema describes a network's meta-structure in addition to its heterogeneous graph definition as: $\mathcal{S} = (\mathcal{R}, \mathcal{O})$. As a result of the network schema, we are able to extract semantic meta-paths between two entities. The following is a formal definition of a meta-path [32]: MP is defined on a network schema and is denoted as a path in the form of $\mathcal{O}_1 \xrightarrow{\mathcal{R}_1} \mathcal{O}_2 \xrightarrow{\mathcal{R}_2} \ldots \xrightarrow{\mathcal{R}_l} \mathcal{O}_{l+1}$. It describes a composite relation $R = R_1 * R_2 * \ldots * R_l$ between object $\mathcal{O}_l$ and $\mathcal{O}_{l+1}$, where $*$ denotes the composition operator on relations.

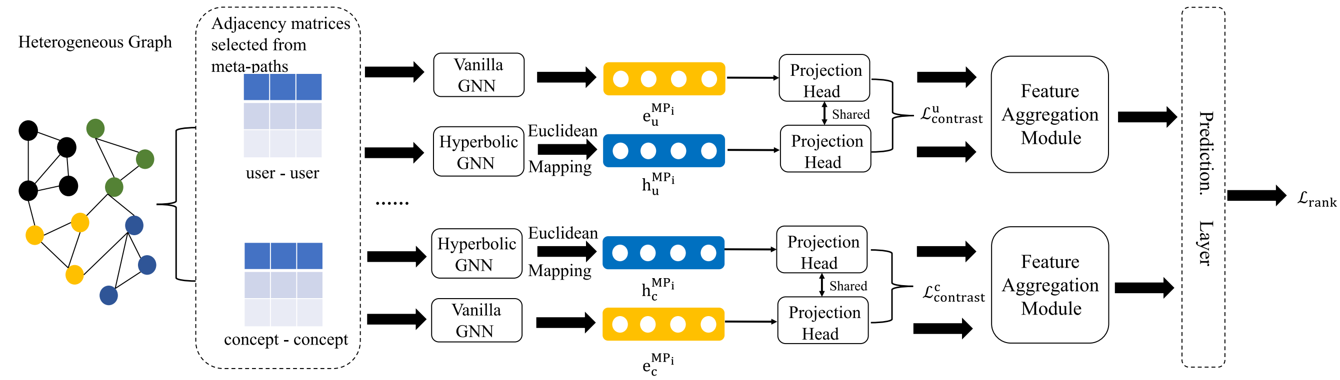

FIGURE 2. **The framework of our proposed approach ROME.**

B. POINCARÉ BALL MODEL

In the presence of constant negative curvature, the Poincaré ball model resembles the Riemannian manifold $(\mathcal{B}, g_x^{\mathcal{B}})$ [33], where $\mathcal{B} = \{x \in \mathbb{R}^n : ||x|| < 1\}$ is an open unit ball. Its metric tensor is $g_x^{\mathcal{B}} = \lambda_x^2 g^E$, where $\lambda_x = \frac{2}{1-||x||^2}$ is the conformal factor and g^E is the Euclidean metric tensor (when $g^E = \mathbf{I}_n$). With a constant negative curvature $-c$, the Poincaré ball model has the following domain:

$$\mathbb{D} = \{(x_1, \ldots, x_n) : x_1^2 + \ldots + x_n^2 < \frac{1}{c}\} \tag{1}$$

where x is a n-dimensional open ball in $\mathbb{R}^n$. Geodesics in the Poincaré ball correspond to straight lines in Euclidean space. As can be seen, the hyperbolic metric tensor is conformal to the Euclidean metric tensor. The induced distance between two points $x, y \in \mathbb{D}^n$ is known to be given by:

$$d_{\mathbb{D}}(x, y) = cosh^{-1}(1 + 2\frac{||x-y||^2}{(1-||x||^2)(1-||y||^2)}) \tag{2}$$

when the constant negative curvature is -1, where $||\cdot||$ is the Euclidean norm and $cosh^{-1}$ is the inverse hyperbolic cosine function. In the Poincaré ball, there is only one center, which is the origin. The origin is regarded as the root node, and leaf nodes spread out layer by layer, capturing information about the tree topology and hierarchical structure of the graph [33].

C. SPACE PROJECTION

Since hyperbolic space has a different gradient than Euclidean space, we cannot directly apply Euclidean gradient optimization methods to it. In other words, existing graph representation learning methods are incompatible with graph representation learning methods in hyperbolic space. The hyperbolic embeddings can be mapped to Euclidean embeddings in order to leverage existing methods for graph learning. In this article, we describe the mechanism by which such mappings operate.

There are two mappings, which are called exponential mapping and logarithmic mapping [14], respectively. For any point $x \in \mathcal{B}$, the mapping from tangent space $\mathcal{T}_x\mathcal{B}_c$ to the hyperbolic space $\mathcal{B}_c$ with the constant negative curvature $-c$ is the exponential mapping, and the mapping from hyperbolic space $\mathcal{B}_c$ to tangent space $\mathcal{T}_x\mathcal{B}_c$ is logarithmic mapping. With $v \in \mathcal{T}_x\mathcal{B}_c$ and $u \in \mathcal{B}_c$, the exponential mapping $exp_x^c : \mathcal{T}_x\mathcal{B}_c \to \mathcal{B}_c$, and the logarithmic mapping $log_x^c : \mathcal{B}_c \to \mathcal{T}_x\mathcal{B}_c$ are defined as:

$$exp_x^c(v) = tanh(\sqrt{c}||v||)\frac{v}{\sqrt{c}||v||} \tag{3}$$

$$log_x^c(u) = artanh(\sqrt{c}||u||)\frac{u}{\sqrt{c}||u||} \tag{4}$$

Hyperbolic embeddings can be mapped to Euclidean space by using the two mapping functions above. A neural network model typically embeds data by multiplying matrixes, adding biases, and activating the layers: $h = \sigma(W \cdot u + b)$. Hyperbolic embeddings cannot be directly processed with weight matrices and biases that are defined in Euclidean space. It is necessary first to translate hyperbolic embeddings into Euclidean embeddings in order to achieve this. For hyperbolic matrix multiplication, we have $W \otimes u = exp_x^c(W \cdot log_x^c(u))$. For hyperbolic bias addition, we have $u \oplus b = exp_x^c(log_x^c(u)+b)$. Thus, the final formalization can be written as:

$$h = exp_x^c(\sigma(log_x^c(W \otimes u \oplus b))) \tag{5}$$

In order to solve graph learning problems in hyperbolic space, we can leverage the transformations mentioned above to utilize the existing graph learning methods in the Euclidean space.

IV. METHOD

In this section, we introduce our proposed approach ROME based on meta-paths in the heterogeneous graph in the MOOCs dataset. We utilize the graph contrastive framework in hyperbolic and Euclidean space to model the matrix factorization for user-user and concept-concept relationships. The final prediction is $\hat{y}_{u,c}$, which denotes the predicted preference score of concept c with respect to user u, respectively. As shown in **Figure 2**, our approach consists of four components, meta-path selection, graph representation, graph contrastive learning, and recommendation prediction.

We will discuss each component in more detail in the following paragraphs.

A. META-PATH SELECTION

It is well known that meta-paths are capable of establishing entity-entity relationships. Similar to previous studies [5], [34], In this paper, we examine user-user and concept-concept relationships in the context of different meta-paths. To fairly compare with [5], [34], our study is based on the same meta-paths used in that study. There are six meta-paths used for MOOCs dataset where four paths for users, including $user \rightarrow concept \rightarrow user$, $user \rightarrow course \rightarrow user$, $user \rightarrow video \rightarrow user$, $user \rightarrow course \rightarrow teacher \rightarrow course \rightarrow user$, and two for concepts including $concept \rightarrow user \rightarrow cpncept$, and $concept \rightarrow course \rightarrow concept$. As a result of extracting a homogeneous graph based on users (concepts), a corresponding adjacency matrix may be created for each meta-path in **Figure 2**. If two users (concepts) are connected via a meta-path, the adjacency matrix entry for that meta-path will be one, otherwise it will be zero. GNNs are then used to learn user representations (concepts) for each meta-path.

B. GRAPH REPRESENTATIONS

As part of our framework, we support two kinds of graph representation and encode the graph structure directly using a neural network model $g(X, A)$, where X is a matrix of node feature vectors and A is the adjacency matrix derived above. The graph encoder includes a vanilla GNN g_E on the common Euclidean space and a hyperbolic GNN g_H based on the hyperbolic space. In terms of the graph encoder, we opt for simplicity by adopting the commonly used graph convolution network (GCN) [35], which is capable of learning the node representation of a graph by inspecting its neighbors. In order to learn user (concept) representations in relation to a meta-path, we apply the linear formulation of layer-wise propagation rule. We initialize $Z^{(0)} = X$ if each node has a set of features, or we can initialize and learn afterward. It can be formulated as follows:

$$Z^{(l+1)} = \sigma(\widetilde{D}^{-\frac{1}{2}}\widetilde{A}\widetilde{D}^{-\frac{1}{2}}Z^{(l)}W^{(l)}) \tag{6}$$

Here, $\widetilde{A} = A + I_N$ is the adjacency matrix of the graph $\mathcal{G}$ with added self-connections. I_N is the identity matrix, $D_{ii} = \sum_j \widetilde{A}_{ij}$ and $W^(l)$ is a layer-specific trainable weight matrix.$\sigma(\cdot)$ denotes an activation function, such as the ReLU. $Z^{(l)}$ is the matrix of activation in the lth layer. User (concept) representations can be derived from the output representation of the last layer. For example, when $l = 2$, the representation in Euclidean space of a user u for a meta-path MP_i will be $e_u^{MP_i}$, which is the output of the last layer of g_E for the meta-path MP_i with respect to u.

We can represent the embeddings in Euclidean space and hyperbolic space for user, respectively, as follows:

$$e_u^{MP_i} = g_E(X, A) \tag{7}$$

$$h_u^{MP_i} = exp_x^c(g_H(X, A)) \tag{8}$$

And the embeddings in Euclidean space and hyperbolic space for concept is derived as $e_c^{MP_i}$ and $h_c^{MP_i}$, respectively

As a result of their strong representative ability, graph embeddings are crucial to the whole framework. In the next step, we will feed these components to the following components.

In hierarchical, taxonomic, and entailment data, hyperbolic representations are superior to Euclidean embeddings. As a result of the negative curvature of the embedding space, disconnected subtrees from the latent hierarchical structure disentangle and cluster in the embedding space. Therefore, sufficient semantics information can be acquired from contrasting pairs in different spaces. In order to achieve this advantage, we propose to construct contrastive learning by constructing pairs from different spaces. Our goal is to obtain informative semantics from different views of the input graph in different spaces by conducting graph contrastive learning between Euclidean embeddings and hyperbolic embeddings. A direct comparison of embeddings in different spaces is not possible without mapping one of them to another space in which a different embedding exists. For a comparison of the Euclidean embedding to the hyperbolic embedding, we must first translate the graph embedding in Euclidean space into hyperbolic space using an exponential function based on knowledge in Section III:

$$e \rightarrow h_u^{MP_i} = exp_x^c(h_u^{MP_i}) \tag{9}$$

The representations are then fed into a shared projection head f, which is an MLP with two hidden layers and PReLU non-linearity, resulting in the final graph representations $h_u^{MP_i}$ and $e \rightarrow h_u^{MP_i}$ for user, and $h_c^{MP_i}$ and $e \rightarrow h_c^{MP_i}$ for concept.

C. GRAPH CONTRASTIVE LEARNING

Objective of self-supervised learning phase is graph contrastive learning between graph hyperbolic embedding $h_u^{MP_i}$ and transformed graph Euclidean embedding $e \rightarrow h_u^{MP_i}$ in the hyperbolic space. Traditional contrastive learning is a method of training that attempts to bring representations with similar semantics closer together and dissimilar ones apart. Due to the fact that these two embeddings are different views of the same input graph, they should be similar, which indicates that there should be a small distance between the two embeddings. In spite of this, these representations may not be sufficiently discriminative and not very robust to representations in generalized spaces, such as the hyperbolic space. Therefore, we propose to implement the contrastive loss in the angular space with a margin to make the loss more discriminative. Let us denote angular θ_{ij} as:

$$\theta_{ij}^u = arccos(\frac{{h_u^{MP_i}}^T e \rightarrow h_u^{MP_j}}{||h_u^{MP_i}|| \cdot ||e \rightarrow h_u^{MP_j}||}) \tag{10}$$

The lack of decision margin can lead to incorrect decisions when a small perturbation occurs around the decision boundary. By adding an additive angular margin m between contrastive pairs, we propose a new training objective for graph representation learning between positive pairs $h_u^{MP_i}$ and $e \to h_u^{MP_i}$, which can be formulated as follows:

$$\mathcal{L}_{contrast}^{u} = -log\frac{e^{cos(\theta_{ii*}^{u}+m)/\tau}}{e^{cos(\theta_{ii*}^{u}+m)/\tau}+\sum_{j\neq i}e^{cos(\theta_{ij}^{u})/\tau}} \tag{11}$$

And the contrastive loss for concept is:

$$\mathcal{L}_{contrast}^{c} = -log\frac{e^{cos(\theta_{ii*}^{c}+m)/\tau}}{e^{cos(\theta_{ii*}^{c}+m)/\tau}+\sum_{j\neq i}e^{cos(\theta_{ij}^{c})/\tau}} \tag{12}$$

In the loss function, the decision boundary for $h_u^{MP_i}$ is $\theta_{ii*}^u + m$. By increasing the compactness of graph representations with the same semantics and enlarging the discrepancy between different semantic representations, it further pushed $h_u^{MP_i}$ toward the area where θ_{ii*}^u gets smaller and θ_{ij}^u gets larger, compared to the traditional contrastive loss. The alignment and uniformity properties are two important measures of representation quality related to contrastive learning, which indicate how well the embeddings are distributed uniformly and how close they are to positive pair embeddings. The final contrastive loss can be formulated as:

$$\mathcal{L}_{contrast} = \mathcal{L}_{contrast}^{c} + \mathcal{L}_{contrast}^{u} \tag{13}$$

D. RECOMMENDATION PREDICTION

We first aggregate the user features and the concepts features in the Euclidean space of each meta-path for the final prediction as follows: $e_u = \sum_{j\in|MP|} e_u^{MP_j}$, and $e_c = \sum_{j\in|MP|} e_c^{MP_j}$. By the same token, we aggregate the features in hyperbolic space, $h_u = \sum_{j\in|MP|} h_u^{MP_j}$, and $h_c = \sum_{j\in|MP|} h_c^{MP_j}$. Based on the acquired representations/embeddings of the users and concepts e_u, h_u, and e_c, h_c. By utilizing the matrix factorization framework, it is possible to calculate the preference score of a concept c for a user u as follows:

$$\hat{y}_{u,c} = e_u^T\Theta_e e_c + \alpha \cdot h_u^T\Theta_h h_c \tag{14}$$

where $\hat{y}_{u,c}$ is the preference score, Θ_e and Θ_h are trainable matrices to let user representation in the same space with concept representation. α is a trainable parameter for the trade-off between the prediction scores from two different views.

For customized ranking we use the Bayesian Personalized Ranking (BPR) [36] which is derived from a Bayesian analysis. According to BPR, a learned concept should be ranked higher (with a higher score) than a random one in a list of concepts with which the user has not interacted, which can be expressed as follows:

$$\mathcal{L}_{rank} = \sum_{u,i,j,i\neq j} -ln(\sigma(\hat{y}_{u,i}-\hat{y}_{u,j})) + \gamma||\Theta||^2 \tag{15}$$

where u, i, j refers to a triplet relationship including a user u, an interacted concept i and an unknown concept j for the user. An interacted concept is preferred over an unknown concept in the former item of the loss, with σ represents a sigmoid active function. γ is the regularization parameter for the L2 norm, and Θ denotes the set of parameters to be learned.

The final loss function of our proposed ROME can be formulated as:

$$\mathcal{L} = \mathcal{L}_{contrast} + \beta\mathcal{L}_{rank} \tag{16}$$

where β is a coefficient for multi-task learning.

V. EXPERIMENTS

A. DATASETS AND IMPLEMENTATION

MOOCCube [37] is an open source large-scale data warehouse that serves research related to MOOCs. In comparison with existing databases of similar educational resources, it is large, rich, and diverse. These records include very detailed information about the student's behavior. This includes the amount of time spent learning, the number of times spent learning, and the intervals between learning videos. A total of nearly 5 million students recorded video viewing for learning purposes, including nearly 200,000 students. As a whole, the dataset consists of 2,005 users, 21,037 concepts, 600 courses, 22,403 videos, 137 schools, 138 teachers, and their relationships. Overall, there are 930,553 interactions between users and concepts, of which 858,072 are in the training set and 72,481 in the test set. **Table 1** presents the overall statistics of the dataset. We selected 706 courses and approximately 40,000 videos for processing, and we can actually use this part of the data to model user behavior and make recommendations in this regard. The final step in establishing the interconnection between the entities, which is the MOOCCube Dataset, is to connect students' behavior and the content of the course with knowledge.

TABLE 1. Details of the MOOCCube dataset for our MOOCs recommendation.

Entities	Statistics	Relations	Statistics
users	2,005	user-concept	930,553
concepts	21,037	user-course	13,696
courses	600	course-video	42,117
videos	22,403	teacher-course	1,875
schools	137	video-concept	295,475
teachers	138	course-concept	150,811

All our experiments are implemented with a single NVIDIA A100 GPU using PyTorch. We optimize the network using mini-batch Adam [38] with momentum. The batch size is set to 1024, and the learning rate is fixed at 0.01. The regularization parameter in Eq. 15 is set as 1e−8, and the dimension of latent features for user (concept) embeddings are set as 30 and 100 respectively as in [5].

B. BASELINE

To demonstrate the effectiveness of ROME, we compare it with several state-of-the-art methods including

- **CMF** [39] is a method for decomposing the data matrix of multiple behavior types at the same time.

- **Metapath2vec** [40] combines meta-path-based random walks with heterogeneous skip-gram models to construct the heterogeneous neighborhood of a node.
- **NMTR** [26] utilizes a joint optimization technique based on multitask learning to combine the advances of NCF modeling.
- **EHCF** [27] is a model that models fine-grained user-item relationships and efficiently learns model parameters without creating negative samples.
- **ACKRec** [28] treats the problem as a rating prediction problem where the rating of a concept is determined by the number of interactions between the user and the concept.
- **LightGCN** [30] simplifies the design of GCN to make it easier to understand and more appropriate for recommendation purposes.
- **MOOCIR** [5] predicts a user's choice of concepts based on those learned user and concept representations.

C. EVALUATION METRICS

The following widely used evaluation metrics are used to evaluate top-k predictions of concepts for users: k is set to 5, 10, 20, and 50. Using the test set, we calculate all metrics for each set of 100 concepts (with one interacted and 99 unknown). We generate a recommendation list for each interacted concept with respect to a user u as: $R_u = \{r_u^1, r_u^2, \cdots r_u^k\}$, where r_u^i indicates concept ranked at the i th position in R_u based on the predicted scores of those concepts.

We apply three widely used metrics, Hit Ratio (HR) [41], Normalized Discounted Cumulative Gain (NDCG) [42], and Mean Reciprocal Rank (MRR) [43] are utilized to evaluate the performance of each method. HR@k represents the proportion of relevant concepts in the test set that are within the top-k concepts of the recommendations as follows:

$$HR@k = \frac{1}{N}\sum_{u}\mathbb{I}(|R_u \cap T_u|) \tag{17}$$

where N is the scale of testing dataset, $\mathbb{I}(x)$ is an indicator function which equals one if $x > 0$ and equals zero otherwise. Based on the rank positions of the relevant concepts, NDCG@k can be calculated as follows:

$$NDCG@k = \frac{1}{Z}\sum_{j=1}^{k}\frac{2^{\mathbb{I}(|R_u^j \cap T_u|)} - 1}{log_2(j+1)} \tag{18}$$

where Z denotes the score obtained by an ideal top-k ranking which serves as a normalization factor. MRR is the average of the reciprocal ranks of positive concepts when they are expressed in reciprocal ranks, which can be expressed as follows:

$$MRR = \frac{1}{N}\sum_{1}^{N}\frac{1}{rank_i} \tag{19}$$

It is important to note that $rank_i$ refers to the position in the corresponding set of 100 concepts of the concept that was interacted with the rest of the unknown concepts. As a recall-based metric, HR measures whether the testing item is in the top-k list, while NDCG is sensitive to position, which assigns a higher score to hits at a higher position. It's noticed that for a user, our evaluation protocol ranks all unobserved items in the training set and thus the obtained results are more persuasive than ranking a random sampling subset.

D. RESULTS

Table 2 reports the HR, NDCG, and MRR scores of the proposed ROME and the compared baselines. From the table above, we have the following observations. **First**, compared to similarity-based recommending methods (like CMF, NMTR and EHCF), graph-based recommending methods (such as LightGCN and MOOCIR) generally outperform similarity-based methods. As a result, we infer that efficient exploitation of graph semantic information can result in discriminative abilities. **Second**, almost all of the evaluation metrics show that our proposed ROME performs significantly better than other baselines. Take the Hit Ratio for example, ROME shows comparatively high scores at 0.855 and 0.948 when $k = 10, 50$, which is 2.27%, and 0.96% over the best baseline. In regard to NDCG, ROME achieves the obviously highest score when $k = 5, 10, 20, 50$ with the improvements of 3.65%, 4.45%, 1.37%, and 2.83%, respectively. As for MRR, ROME still achieves the obviously highest score 0.497, which is 2.69% over the best baseline MOOCIR. Our ROME is effective in two ways. (i) Introduction of our multi-view contrastive learning on graph representations, which makes full use of both capacities of Euclidean and hyperbolic space. (ii) Introduction of contrastive loss on angular space, which further increases the model discrimination ability, which is further studied in the part below.

E. ABLATION STUDY

During this part, we examine four variants of our proposed ROME in order to determine the effectiveness of different components: (1) **ROME\M** removes the multi-view framework by cutting off the branch of hyperbolic representations and removes the contrastive learning. (2) **ROME\A** replaces the contrastive loss in angular space with a margin by the simple InfoNCE [12] loss. (3), **ROME\V** replaces the hyperbolic representations in the multi-view framework with the representations from the vanilla model in Euclidean space. (4), **ROME\P** cuts off the feature learning of $h_u^T \Theta_h h_c$ from hyperbolic space. (5), **ROME\MA** combines the operations in **ROME \M** and **ROME \A** together; **ROME \VA** combines the operations in **ROME \V** and **ROME \A** together; **ROME \AP** combines the operations in **ROME \A** and **ROME \P** together. Results of our ROME and its variants under different evaluation settings are shown in **Table 3**.

As a result of the results, we can draw the following conclusions: (1) The multi-view representations for graph contrastive learning adds extra semantic information for

TABLE 2. The recommendation accuracies of different methods measured by HR, NDCG and MRR.

Methods	HR@5	HR@10	HR@20	HR@50	NDCG@5	NDCG@10	NDCG@20	NDCG@50	MRR
CMF	0.620	0.741	0.828	0.859	0.432	0.479	0.499	0.509	0.418
Metapath2vec	0.642	0.773	0.873	0.906	0.468	0.511	0.537	0.582	0.440
NMTR	0.651	0.772	0.829	0.894	0.470	0.516	0.542	0.591	0.449
EHCF	0.649	0.768	0.833	0.894	0.475	0.521	0.547	0.593	0.462
ACKRec	0.659	0.764	0.842	0.881	0.503	0.538	0.557	0.601	0.475
LightGCN	0.688	0.759	0.851	0.894	0.515	0.548	0.569	0.617	0.477
MOOCIR	0.704	0.836	**0.922**	0.939	0.520	0.562	0.584	0.637	0.484
ROME (Ours)	**0.710**	**0.855**	0.907	**0.948**	**0.539**	**0.587**	**0.592**	**0.655**	**0.497**

TABLE 3. Ablation study of our proposed ROME and its four variants under different evaluation metrics.

Methods	HR@5	HR@10	HR@20	HR@50	NDCG@5	NDCG@10	NDCG@20	NDCG@50	MRR
ROME**M**	0.706	0.839	0.898	0.932	0.527	0.571	0.580	0.649	0.486
ROME**A**	0.708	0.850	0.901	0.939	0.532	0.581	0.585	0.651	0.492
ROME**V**	0.703	0.846	0.898	0.936	0.533	0.584	0.589	0.652	0.492
ROME**P**	0.704	0.841	0.899	0.931	0.525	0.569	0.577	0.650	0.484
ROME	**0.710**	**0.855**	**0.907**	**0.948**	**0.539**	**0.587**	**0.592**	**0.655**	**0.497**

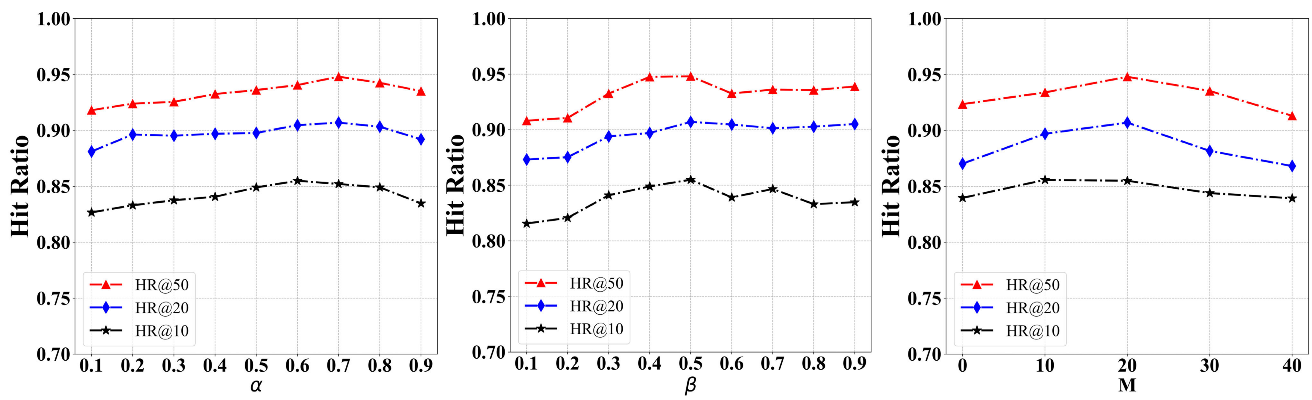

FIGURE 3. Sensitivity analysis of three important hyper-parameters.

representations and solves the challenges to the recommendation system indeed. Specifically, the removal of hyperbolic representations causes a drastic decrease on average on each evaluation. (2) Our method outperforms **ROME\A** about 0.59%, which demonstrates the effectiveness of the further mapping on the angular space to make the contrastive loss more discriminative based on the representations on hyperbolic representations. (3) After considering the hyperbolic space for multi-view graph representation learning, our method has gained improvement over **ROME\V**, indicating the importance of exploring the geometry view when learning discriminative features. (4) The feature learning in the hyperbolic space and the contrastive loss in angular space with a margin is key to the performances, especially when removing **ROME \ AP**. Then, replacing hyperbolic space representations with Euclidean representations is worse than removing the representations in the hyperbolic space. It can be attributed that the contrastive loss can not be distinguished by the representations from the same space for the multi-view framework.

F. SENSITIVITY ANALYSIS

In this part, we study the impacts of three hyper-parameters, including two balance coefficients α and β in Eq. 14 and Eq. 16, and the margin of angular space m in Eq. 11 and Eq. 12 respectively by conducting experiments evaluated by HR. We first vary α from 0.1 to 0.9 with other parameters fixed, which controls the influence of the representations from the hyperbolic space. As it is shown in Fig. 3, we can observe that when α is small, a higher coefficient makes the recommendation results better, which improves the representation ability of our ROME. We find that the performance of our method is not sensitive to α in the range of $[0.6, 0.7]$. Then, we fix all other hyper-parameters and vary β from 0.1 to 0.9, which decides the coefficient between two different optimization target. According to the results of the sensitivity study, the threshold is not sensitive when β is around 0.5. As for the margin of the angular space, we vary M with the stride 10 from 0 degree to 40. It is obvious that the performances are stable in the range [10, 20], which directly decides the discriminative ability. Towards the end, we set α, β and M to 0.8, 0.3 and 20, respectively.

VI. CONCLUSION

The paper investigates the problem of online MOOCs recommendation and present a novel approach named ROME which simultaneously learn graph representations from two views to solve this problem. We not only utilize the representations from vanilla GNNs in Euclidean space, but also utilize the GNNs model in hyperbolic space as extraordinary semantic information. Then we integrate the both views by mapping it to the same hyperbolic space to implement contrastive learning framework with angular margin for more discriminative target. Extensive experiments on popular MOOCCube datasets verify the superiority of our ROME compared with a range of state-of-the-art methods. In future work, we will extend our proposed ROME to a range of relevant fields such as semi-supervised recommendation and more real-worlf scenes.

REFERENCES

[1] Y. You, T. Chen, Y. Sui, T. Chen, Z. Wang, and Y. Shen, ''Graph contrastive learning with augmentations,'' in *Proc. NIPS*, vol. 33, 2020, pp. 5812–5823.

[2] Z. Wang, W. Wei, G. Cong, X.-L. Li, X.-L. Mao, and M. Qiu, ''Global context enhanced graph neural networks for session-based recommendation,'' in *Proc. 43rd Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jul. 2020, pp. 169–178.

[3] C. Lei, Y. Liu, L. Zhang, G. Wang, H. Tang, H. Li, and C. Miao, ''SEMI: A sequential multi-modal information transfer network for E-commerce micro-video recommendations,'' in *Proc. 27th ACM SIGKDD Conf. Knowl. Discovery Data Mining*, Aug. 2021, pp. 3161–3171.

[4] D. Wu, X. Luo, Z. Ma, C. Chen, M. Deng, and J. Ma, ''ARGO: Modeling heterogeneity in E-commerce recommendation,'' in *Proc. Int. Joint Conf. Neural Netw. (IJCNN)*, Jul. 2021, pp. 1–8.

[5] G. Piao, ''Recommending knowledge concepts on MOOC platforms with meta-path-based representation learning,'' in *Proc. 14th Int. Conf. Educ. Data Mining*, 2021, pp. 1–8.

[6] P. Hao, Y. Li, and C. Bai, ''Meta-relationship for course recommendation in MOOCS,'' *Multimedia Syst.*, vol. 29, pp. 1–12, Aug. 2022.

[7] T. Ebesu, B. Shen, and Y. Fang, ''Collaborative memory network for recommendation systems,'' in *Proc. 41st Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jun. 2018, pp. 515–524.

[8] X. He, L. Liao, H. Zhang, L. Nie, X. Hu, and T.-S. Chua, ''Neural collaborative filtering,'' in *Proc. 26th Int. Conf. World Wide Web*, 2017, pp. 173–182.

[9] X. He, Z. He, J. Song, Z. Liu, Y.-G. Jiang, and T.-S. Chua, ''NAIS: Neural attentive item similarity model for recommendation,'' *IEEE Trans. Knowl. Data Eng.*, vol. 30, no. 12, pp. 2354–2366, Dec. 2018.

[10] F. Xue, X. He, X. Wang, J. Xu, K. Liu, and R. Hong, ''Deep item-based collaborative filtering for Top-N recommendation,'' *ACM Trans. Inf. Syst.*, vol. 37, no. 3, pp. 1–25, Jul. 2019.

[11] J. Yu, H. Yin, X. Xia, T. Chen, L. Cui, and Q. V. H. Nguyen, ''Are graph augmentations necessary? Simple graph contrastive learning for recommendation,'' in *Proc. 45th Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jul. 2022, pp. 1294–1303.

[12] A. Van Den Oord, Y. Li, and O. Vinyals, ''Representation learning with contrastive predictive coding,'' 2018, *arXiv:1807.03748*.

[13] J. W. Cannon, W. J. Floyd, R. Kenyon, and W. R. Parry, ''Hyperbolic geometry,'' *Flavors geometry*, vol. 31, nos. 59–115, p. 2, 1997.

[14] I. Chami, Z. Ying, C. Ré, and J. Leskovec, ''Hyperbolic graph convolutional neural networks,'' in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 32, 2019, pp. 1–12.

[15] Y. Zhang, R. He, Z. Liu, K. H. Lim, and L. Bing, ''An unsupervised sentence embedding method by mutual information maximization,'' in *Proc. Conf. Empirical Methods Natural Lang. Process. (EMNLP)*, 2020, pp. 1–10.

[16] J. Giorgi, O. Nitski, B. Wang, and G. Bader, ''DeCLUTR: Deep contrastive learning for unsupervised textual representations,'' 2020, *arXiv:2006.03659*.

[17] J. Yu, H. Yin, X. Xia, T. Chen, J. Li, and Z. Huang, ''Self-supervised learning for recommender systems: A survey,'' 2022, *arXiv:2203.15876*.

[18] C. Gao, X. Wang, X. He, and Y. Li, ''Graph neural networks for recommender system,'' in *Proc. 15th ACM Int. Conf. Web Search Data Mining*, Feb. 2022, pp. 1623–1625.

[19] P. Velickovic, W. Fedus, W. L. Hamilton, P. Liò, Y. Bengio, and R. D. Hjelm, ''Deep graph infomax,'' in *Proc. ICLR*, vol. 2, no. 3, 2019, p. 4.

[20] F.-Y. Sun, J. Hoffmann, V. Verma, and J. Tang, ''InfoGraph: Unsupervised and semi-supervised graph-level representation learning via mutual information maximization,'' 2019, *arXiv:1908.01000*.

[21] R. D. Hjelm, A. Fedorov, S. Lavoie-Marchildon, K. Grewal, P. Bachman, A. Trischler, and Y. Bengio, ''Learning deep representations by mutual information estimation and maximization,'' 2018, *arXiv:1808.06670*.

[22] K. Hassani and A. H. Khasahmadi, ''Contrastive multi-view representation learning on graphs,'' in *Proc. Int. Conf. Mach. Learn.*, 2020, pp. 4116–4126.

[23] R. I. Kondor and J. Lafferty, ''Diffusion kernels on graphs and other discrete structures,'' in *Proc. 19th Int. Conf. Mach. Learn.*, 2002, pp. 315–322.

[24] L. Page, S. Brin, R. Motwani, and T. Winograd, ''The PageRank citation ranking: Bringing order to the web,'' Stanford InfoLab, Stanford, CA, USA, Tech. Rep., 1999.

[25] J. Qiu, Q. Chen, Y. Dong, J. Zhang, H. Yang, M. Ding, K. Wang, and J. Tang, ''GCC: Graph contrastive coding for graph neural network pre-training,'' in *Proc. 26th ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2020, pp. 1150–1160.

[26] C. Gao, X. He, D. Gan, X. Chen, F. Feng, Y. Li, T.-S. Chua, and D. Jin, ''Neural multi-task recommendation from multi-behavior data,'' in *Proc. IEEE 35th Int. Conf. Data Eng. (ICDE)*, Apr. 2019, pp. 1554–1557.

[27] C. Chen, M. Zhang, Y. Zhang, W. Ma, Y. Liu, and S. Ma, ''Efficient heterogeneous collaborative filtering without negative sampling for recommendation,'' in *Proc. AAAI Conf. Artif. Intell.*, vol. 34, no. 1, 2020, pp. 19–26.

[28] J. Gong, S. Wang, J. Wang, W. Feng, H. Peng, J. Tang, and P. S. Yu, ''Attentional graph convolutional networks for knowledge concept recommendation in MOOCs in a heterogeneous view,'' in *Proc. 43rd Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jul. 2020, pp. 79–88.

[29] F. Wu, A. Souza, T. Zhang, C. Fifty, T. Yu, and K. Weinberger, ''Simplifying graph convolutional networks,'' in *Proc. Int. Conf. Mach. Learn.*, 2019, pp. 6861–6871.

[30] X. He, K. Deng, X. Wang, Y. Li, Y. Zhang, and M. Wang, ''LightGCN: Simplifying and powering graph convolution network for recommendation,'' in *Proc. 43rd Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jul. 2020, pp. 639–648.

[31] Y. Sun and J. Han, ''Mining heterogeneous information networks: A structural analysis approach,'' *ACM SIGKDD Explor. Newslett.*, vol. 14, no. 2, pp. 20–28, 2013.

[32] Y. Sun, J. Han, X. Yan, P. S. Yu, and T. Wu, ''PathSim: Meta path-based top-k similarity search in heterogeneous information networks,'' *Proc. VLDB Endowment*, vol. 4, no. 11, pp. 992–1003, Aug. 2011.

[33] O. Ganea, G. Bécigneul, and T. Hofmann, ''Hyperbolic neural networks,'' in *Proc. Adv. Neural Inf. Process. Syst.*, vol. 31, 2018, pp. 1–11.

[34] C. Shi, B. Hu, W. X. Zhao, and P. S. Yu, ''Heterogeneous information network embedding for recommendation,'' *IEEE Trans. Knowl. Data Eng.*, vol. 31, no. 2, pp. 357–370, Feb. 2019.

[35] T. N. Kipf and M. Welling, ''Semi-supervised classification with graph convolutional networks,'' 2016, *arXiv:1609.02907*.

[36] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme, ''BPR: Bayesian personalized ranking from implicit feedback,'' 2012, *arXiv:1205.2618*.

[37] J. Yu, G. Luo, T. Xiao, Q. Zhong, Y. Wang, W. Feng, J. Luo, C. Wang, L. Hou, J. Li, Z. Liu, and J. Tang, "MOOCCube: A large-scale data repository for NLP applications in MOOCs," in *Proc. 58th Annu. Meeting Assoc. Comput. Linguistics*, 2020, pp. 3135–3142.
[38] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," 2014, *arXiv:1412.6980*.
[39] Z. Zhao, Z. Cheng, L. Hong, and E. H. Chi, "Improving user topic interest profiles by behavior factorization," in *Proc. 24th Int. Conf. World Wide Web*, May 2015, pp. 1406–1416.
[40] Y. Dong, N. V. Chawla, and A. Swami, "metapath2vec: Scalable representation learning for heterogeneous networks," in *Proc. 23rd ACM SIGKDD Int. Conf. Knowl. Discovery Data Mining*, Aug. 2017, pp. 135–144.
[41] G. Karypis, "Evaluation of item-based top-*N* recommendation algorithms," in *Proc. 10th Int. Conf. Inf. Knowl. Manage.*, 2001, pp. 247–254.
[42] K. Järvelin and J. Kekäläinen, "IR evaluation methods for retrieving highly relevant documents," in *Proc. 23rd Annu. Int. ACM SIGIR Conf. Res. Develop. Inf. Retr.*, Jul. 2000, pp. 243–250.
[43] N. Craswell, "Mean reciprocal rank," *Encyclopedia Database Syst.*, vol. 1703, 2009.

HAO LUO received the master's degree from Yunnan University, Yunnan, China. He is currently pursuing the Ph.D. degree in intelligent systems with Universiti Putra Malaysia. He not only has many years of scientific research and teaching experience, but also a lot of engineering practice experience, and he is good at machine learning and deep learning algorithms. His current research interest includes the application of recommendation system for the educational field.

NOR AZURA HUSIN received the Ph.D. degree in artificial intelligent from Universiti Putra Malaysia, Malaysia, in 2013. She is currently a Senior Lecturer with the Department of Computer Science, Faculty of Computer Science and Information Technology, Universiti Putra Malaysia. Her expertise is in the development of prediction models, and she has mainly contributed to the fields of medicine and agriculture for dengue outbreak prediction and paddy disease detection. She worked on a consulting project for the design of a web-based and mobile-based online logistics aggregation business. She has also served as a Consultant with the Lembaga Hasil Dalam Negeri Malaysia (LHDNM) on an efficacy study about the implementation of subsistence allowance. Her research interests include data mining, machine learning, and intelligent computing.

TEH NORANIS MOHD ARIS received the bachelor's degree (Hons.) in information technology from Universiti Utara Malaysia (UUM), in 1994, the Master of Science degree in artificial intelligence from Universiti Putra Malaysia (UPM), in 2002, and the Ph.D. degree from Universiti Kebangsaan Malaysia (UKM), in 2008. She is currently an Associate Professor with the Faculty of Computer Science and Information Technology, UPM. Her research interests include artificial intelligence, fuzzy systems, image processing, software agent, and programming science.