Dual-View Fusion of Heterogeneous Information Network Embedding for Recommendation

Jinlong Ma, and Runfeng Wang

***Abstract*—Heterogeneous Information Networks (HINs) contain rich semantic information due to their involvement of multiple types of nodes and edges. Heterogeneous network embedding is used to analyze HINs by embedding network information in low-dimensional node representations. However, existing heterogeneous embedding methods either ignore the implicit topological relationships between distant nodes or neglect nodes features and meta-paths information disparities, which reflects that extracting HIN embeddings from a single view may lead to incomplete information extraction. In order to make the information extraction more complete, we propose a dual-view fusion heterogeneous information network embedding method (DFHE) for recommendation tasks. Specifically, it extracts effective features from HINs from both the remote topology view and the semantic aggregation view: the remote topology view uses a meta-graph-guided random walk to capture the topological relationships between remote nodes and learns embeddings through a graph convolutional network (GCN) encoder, while the semantic aggregation view uses an attention mechanism to learn the importance of different meta-paths, node relationships, and aggregates the semantic information of each meta-path. Experimental results on two real-world network datasets demonstrate an enhancement in recommendation task performance under the application of DFHE, compared to the baseline. This improvement persists even when some meta-paths are deleted, thereby verifying the method's effectiveness.**

Link to graphical and video abstracts, and to code: https://latamt.ieeer9.org/index.php/transactions/article/view/8856

***Index Terms*—Heterogeneous Information Network, Network Embedding, Attention Mechanism, Recommender System.**

I. Introduction

In the process of information network embedding, the representations of nodes and edges within a network are transformed from a high-dimensional sparse vector space to a low-dimensional dense vector space, thereby showcasing its effectiveness and adaptability. Network embedding not only tackles problems inherent in network data (such as data sparsity [1]) but also achieves significant results in various downstream tasks, including node classification [2], [3], link prediction [4], node clustering [1], [5], network visualization [6], and recommendation systems [7]. Depending on the nature of the network, we can categorize information network embedding into two types: homogeneous and heterogeneous information network embedding.

The embedding of homogeneous information networks has been widely applied, including well-established methods such as DeepWalk [8], LINE [9], and Node2Vec [10]. However, the diversification of data sources in the real world and the intricacy of data structures have resulted in the emergence of heterogeneous information networks (HINs) through various data interactions. HINs are widely used in many real-world situations. These include social media platforms, knowledge bases and academic reference networks. HINs encompass a variety of entities and connections. For example, the HIN of the Douban movie dataset shown in Fig. 1a features five different node types and multiple relationships among them. It becomes imperative that HINs encapsulate more intricate structural details and a broader semantic spectrum in contrast to homogeneous networks, which are characterized by a uniform type of nodes and edges. Employing methods designed for homogeneous information network embedding on HINs may result in partial information extraction, thereby potentially impacting the efficacy of subsequent tasks. Therefore, it is essential to take into account the heterogeneity of nodes and edges, as well as the semantic and structural attributes of the network, to efficiently integrate diverse information in HIN. [11].

A meta-path is a sequential configuration of node and edge types within a network schema, elucidating the complex interrelations among the participating node types. As an efficient way to represent semantic information, meta-paths have been widely used in research [12]–[14]. Drawing from the Deepwalk [8] method used in homogeneous networks, the method of meta-path-guided HIN embedding conducts random walks, which are directed by one or multiple meta-paths on the HIN, and it yields node embedding vectors using the Skip-gram [15] method. However, in real-world scenarios, HINs often have sparse connectivity or a significant number of absent links due to the difficulty of information acquisition. Therefore, methods based on meta-paths may not effectively discern the underlying information between distantly connected nodes. In contrast, the meta-graph [16] guides random walks by capturing richer relationships between remote nodes, providing a more flexible matching method when representing complex structures between nodes.

Meanwhile, the rapid advancement of deep learning has led to the emergence of Graph Neural Networks (GNNs) [17]–[19], which learn the network's representation through neural layer architecture and have shown superior effectiveness in various subsequent tasks. Most of the existing heterogeneous graph neural networks learn HIN representations by sampling neighboring nodes based on meta-paths and aggregating node information [20], [21]. The diverse nature of the network structure and node content

J. L. Ma, and R. F. Wang are with Hebei University of Science and Technology, Shijiazhuang, Hebei, China (e-mails: mzjinlong@163.com, and 1293221297@qq.com).

within HINs poses a significant challenge for GNNs in their endeavor to encapsulate this rich and varied information into low-dimensional vector spaces.

To fully extract valid information from HIN, many researchers have performed information fusion from different views of HIN. HERec [22] introduces three fusion methods to merge embeddings of nodes across various meta-paths, thereby showcasing the efficiency of nonlinear fusion methods. MEOW [21] constructs two views: a coarse-grained view is used to reflect the connection of objects on the meta-path, while a fine-grained view describes the details of object connections through the meta-path context. MAGNN [19] employs an attention mechanism for both intra and inter-meta-path aggregations, integrating this data into the ultimate node embedding, thus capturing the extensive semantics within the HIN. MFHE [11] adeptly captures local information within sub-views utilizing a multi-head attention mechanism, subsequently amalgamating the sub-view information via a spatial matrix. These studies show that the method of collating network information from diverse perspectives is indeed effective.

To tackle the aforementioned problems, a method called dual-view fusion heterogeneous information network embedding for recommendation (DFHE) is proposed. DFHE captures the effective information of HIN separately through the remote topology layer and the semantic aggregation layer and finally fuses this information to generate the final node embedding. Specifically, at the remote topology layer, DFHE guides random walks through meta-graphs to adequately encode topological relationships between remote nodes at the network layer and then processes the walk data through GCNs to obtain network topological information embeddings. Through the semantic aggregation layer, DFHE employs an attention strategy to compile internal data within the network, guided by meta-paths. It then further utilizes an attention strategy to allocate weights to paths and carry out aggregation across meta-paths, thereby securing the embeddings of semantic data within the network. Finally, the obtained embeddings are fused again to obtain the rich synthesized semantics in HIN.

As follows is a summary of the paper's key contributions:

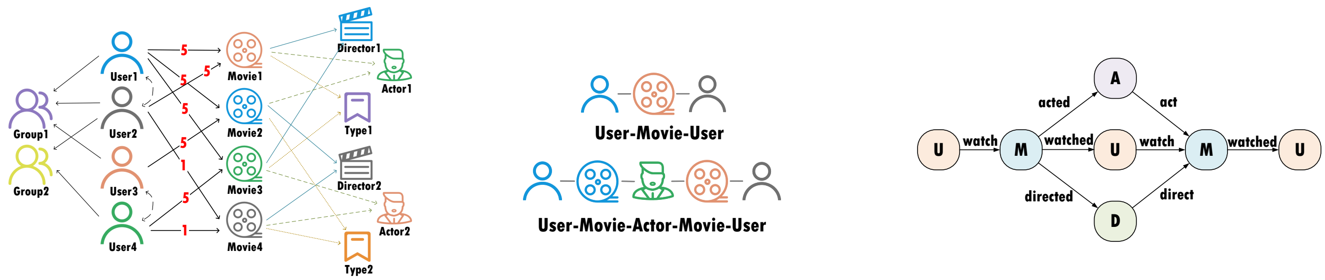
1) A new method DFHE is proposed, which consists of three main parts: remote topology layer, semantic aggregation layer, dual-view information fusion and recommendation. The DFHE method fully takes into account the complexity of semantics and structure in heterogeneous networks and thus is suitable for complex HIN information extraction tasks.
2) To confirm the efficacy of the DFHE method, we perform experimental tasks of recommendation on two datasets, Douban Movie and Yelp. The experimental outcomes demonstrate the proficiency of the DFHE method in enhancing the recommendation performance of HINs.
3) We also experimentally performed a meta-path sensitivity analysis and a comparison of the efficiency with mainstream methods.

II. Related Works

In initial studies, methods on network embedding were mostly developed around homogeneous networks. Among them, network embedding grounded on random walks has gained widespread acceptance and application. The fundamental principle is to capture the network nodes' embedded depiction through the study of their respective neighborhoods' embedded depictions. Influenced by the classic Word2vec [23] method in natural language processing, Deepwalk [8] employs a strategy that randomly chooses walk paths, treating these paths as sentences and the nodes within them as words. Meanwhile, the LINE [9] method focuses on key features. It takes into account the second-order similarity of nodes, which makes it more advantageous when dealing with complex network structures. Beyond methods that utilize random walk strategies, SDNE [24] has also devised a vector representation for automated dimension reduction. This depiction grasps the complex non-linear framework of the network via the application of multi-layer non-linear functions.

The process of HIN embedding necessitates additional procedures on various kinds of nodes and edges. Metapath2vec [12] produces random walks steered by a solitary meta-path and supplies them to the skip-gram [15] for the creation of node embeddings. HERec [22] converts the heterogeneous network into a homogeneous one, utilizing meta-path neighbors. It employs the Deepwalk method to derive embeddings for distinct node types and designs various fusion functions for information integration. HetNERec [14], based on HERec, builds a co-occurrence network through network co-occurrence relationships and achieves good results. HopRec [13] replicates the pairwise interaction of user and item embeddings through the application of the outer product. McRec [37] uses a meta-path context for top-N recommendations, needing only user-to-item meta-paths. RESCHet [25] innovatively incorporates the embedding spectral clustering method into the HIN recommendation system. Furthermore, RMS-HRec [26] establishes a reinforcement learning framework and instructs a policy network to autonomously formulate relationships, thereby discovering significant meta-paths. These methods provide new perspectives and methods for HIN embedding research.

Within the realm of GNN, deep neural networks coupled with message-passing mechanisms are employed for the learning of node embeddings. GraphGAN [27] leverages the principles of spatial generative adversarial networks for network embedding, utilizing the attribute data of vertices to depict unidentified vertices throughout the training process. The GCN [28] first proposed a convolutional method to integrate network structural features, providing an effective perspective for research. The Graph Attention Network (GAT) [17] method incorporates a multi-head attention mechanism within its design. This method is integrated into the aggregation function to evaluate the significance of information from each neighboring node as viewed by the target node. Nonetheless, all aforementioned methods are tailored for isomorphic information networks and are not



(a) HIN (b) Meta-path (c) Meta-graph (partial)

Fig. 1. Example of HIN for Douban movie dataset; (b) and (c) are examples of the meta-paths and meta-graph corresponding to HIN (a).

directly applicable for HIN embedding. To address HIN, a number of heterogeneous GNNs have been put forth. HAN [18] converts a heterogeneous network into a cohesive network connected via multiple meta-paths, aggregating information from neighbors determined by these meta-paths using a graph attention network framework, and consolidates different meta-paths through an attention mechanism. MAGNN [19] utilizes aggregation within meta-paths to merge embeddings from intermediate nodes, and across meta-paths to integrate messages from various paths. HHSE [20] utilizes residual attention to enhance the information representation of hidden layer nodes. Regarding attribute complementation, Jin $et\,al.$ put forward HGNN-AC [29], a proposal aimed at supplementing the attribute details of HINs.

Meanwhile, in the field of recommendation algorithms we apply, non-meta-path based methods are also widely used, such as MF-based methods, regularisation-based methods and GNN-based methods. BPR [31] uses users' implicit feedback for recommendations. NCF [32] uses neural networks to capture user-item interactions and improve user interest and preference understanding. CKE [33] uses structural, textual, and visual data from knowledge bases to understand item semantics and derive embeddings. CFKG [34] expands traditional collaborative filtering to learn heterogeneous knowledge base embedding. GEMS [35] uses a concurrent genetic method to identify relevant meta-structures for recommendations. KGAT [36] models high-order relationships within a graph neural network. Most non-meta-path based approaches rely on attribute information of nodes and may not fully exploit the potential connection between users and items on HIN for recommendation.

Nevertheless, the methods we discussed earlier tend to either neglect the topological data from distant nodes within the network or ignore the correlation among different views. As a result, there is potential for enhancing the preservation of semantic information.

III. Preliminary

In this section, we will present an array of definitions and corresponding instances pertaining to the embedding of HINs.

Definition 1: Heterogeneous Information Network. A Heterogeneous Information Network (HIN) is defined as a network that comprises diverse categories of nodes and edges. The information network is represented as $G = (V, E, A, R)$. V represents the collection of nodes, E represents the collection of relations, A represents the collection of node types and R represents the collection of relation types. The node type mapping function $\phi : V \to A$ and the relation type mapping function $\psi : E \to R$. An information network is classified as a HIN if it encompasses various node types and relationships, as stipulated by the condition $|A|+|R| > 1$. For example, Fig. 1a illustrates a HIN instance that includes six types of nodes and six types of relationships.

Definition 2: Meta-schema. Given a HIN $G = (V, E, A, R)$ in Definition 1, the meta-schema of a network G can be represented as $T_G = (A, R)$. Here, A and R serve as the nodes and relationships respectively, forming a graph that represents the node types and relation types. This is shown in Fig. 3.

Definition 3: Meta-path. A meta-path P is a path defined on the graph of the meta-schema $T_G = (A, R)$, which can be of the form $A_1 \xrightarrow{R_1} A_2 \xrightarrow{R_2} \cdots \xrightarrow{R_l} A_{l+1}$, and is a sequence of nodes. It describes the composite relation $R_1 \circ R_2 \circ \cdots \circ R_l$ between nodes A_1 and A_{l+1}. Here, $\circ$ is the relation combination operator, which combines the individual relations $R_1, R_2, \cdots, R_l$ to form a composite relation between the nodes. Two examples of meta-paths are shown in Fig. 1b: 'User-Movie-User' and 'User-Movie-Actor-Movie-User', where the first meta-path indicates that two users jointly watch the same movie, and the second meta-path indicates that both users watched a movie starring the same actor.

Definition 4: Meta-path-based Neighbors. For a node v, its neighbors N_v^p along the meta-path P are the collection of nodes linked to v via meta-path P within the network. Neighbors determined through meta-paths are derived from multiplying sequences of adjacency matrices. For instance, in Fig. 1a, the meta-path 'User-Movie-User' yields User1 and User4 as User2's neighbors.

Definition 5: Meta-graph. A meta-graph, represented as $\mathcal{G} = (N, M, n_s, n_t)$, is a directed acyclic graph constructed based on a specified HIN meta-schema $T_G = (A, R)$. Here, N and M denote the collections of node types and edge types respectively, with $N \in A$ and $M \in R$. Typically, a meta-graph consists of a unique source node n_s and a unique target node n_t. For clear identification, we define the topological sequence of each node in N within $\mathcal{G}$ as its layer, denoted by $d_{\mathcal{G}}$. Based on the layers of the nodes, N can be divided into disjoint sub-layers $N[i]\,(1 \le i \le d_{\mathcal{G}})$, which represent the set of nodes in the i-th layer. Meanwhile, for each node n in N, we denote $l(n) = i$ as the respective layer of node n.

For a meta-graph $\mathcal{G} = (N, M, n_s, n_t)$ where $n_s = n_t$, its recursive meta-graph $\mathcal{G}^\infty = (N^\infty, M^\infty, n_s^\infty, n_t^\infty)$ is constructed by concatenating any number of $\mathcal{G}$ in an end-to-end manner. In the recursive meta-graph $\mathcal{G}$, we designate the layer of each node $n \in N^\infty$ as $l^\infty(n)$.

Considering the meta-graph illustrated in Fig. 1c, one possible random walk might be: User 1 $\rightarrow$ Movie 1 $\rightarrow$ Actor 1 $\rightarrow$ Movie 2 $\rightarrow$ Director 2 $\rightarrow$ Movie 4 $\rightarrow$ User 2. This describes that Actor 1 starred in Movie 1 and Movie 2, Director 2 directed Movie 2 and Movie 4, and both User 1 and User 2 watched the works of Director 2. Compared with the meta-path given in Fig. 1b, the meta-graph $\mathcal{G}$ encapsulates more complex connections among remote nodes.

IV. The Proposed Method

In this section, we will elaborate on the proposed dual-view fusion based HIN embedding method DFHE in detail. As depicted in Fig. 2, the DFHE framework is constructed by three main parts: the remote topology layer, the semantic aggregation layer, and the dual-view information fusion and recommendation.

A. Remote Topology Layer View

In the remote topology layer, we first guide a random walk on the HIN through the meta-graph and input the output walk sequence into a GCN auto-encoder to obtain remote topology feature embeddings. In a HIN $G = (V, E, A, R)$, given the corresponding meta-graph $\mathcal{G} = (N, M, n_s, n_t)$ with $n_s = n_t$, we can obtain the corresponding recursive meta-graph $\mathcal{G}^\infty = (N^\infty, M^\infty, n_s^\infty, n_t^\infty)$. At this time, the node sequence $\mathcal{S}_{\mathcal{G}} = \{v_1, v_2, \cdots, v_L\}$ from the random walk, guided by the meta-graph $\mathcal{G}$, ensures that for every v_i in $\mathcal{S}_{\mathcal{G}}$ with i ranging from 1 to L, $v_i \in V$ and $(v_{i-1}, v_i) \in E$. This indicates the walk's adherence to the network structure. We use $\phi(v_i)$ to represent the corresponding node type.

Starting with a node n_s, the process for a random walk guided by a meta-graph may begin. At the i-th step, the transition probability under the guidance of the meta-graph $\mathcal{G}$ is denoted as $Pr(v_i|v_{i-1}; \mathcal{G}^\infty)$. Here, the transition probability $Pr(v_i|v_{i-1}; \mathcal{G}^\infty)$ becomes zero in two scenarios. The first scenario is when the edge (v_{i-1}, v_i) exists in the set E, but in its recursive meta-graph $\mathcal{G}^\infty$, there is no connection from the node type $\phi(v_{i-1})$ at layer $l^\infty(\phi(v_{i-1}))$ to the node type $\phi(v_i)$. The second scenario is when the edge (v_{i-1}, v_i) does not exist in the set E. Apart from that, $Pr(v_i|v_{i-1}; \mathcal{G}^\infty)$ is defined as follows:

$$Pr(v_i|v_{i-1}; \mathcal{G}^\infty) = \frac{1}{T_{\mathcal{G}^\infty}(v_{i-1})} \times \frac{1}{|\{u|(v_{i-1}, u) \in E, \phi(v_i) = \phi(u)\}|}\,, \quad (1)$$

where $|\{u|(v_{i-1}, u) \in E, \phi(v_i) = \phi(u)\}|$ denotes the count of neighbors that are directly linked to the node v_{i-1} and possess the node type $\phi(v_i)$, and $T_{\mathcal{G}^\infty}(v_{i-1})$ represents the count of edges that originate from v_{i-1} and comply with the constraints of the recursive meta-graph $\mathcal{G}$. It can be expressed as follows:

$$T_{\mathcal{G}^\infty}(v_{i-1}) = |\{j|(\phi(v_{i-1}), \phi(u)) \in M^\infty \bigcap (N^\infty[l^\infty(\phi(v_{i-1}))] \times N^\infty[j]), (v_{i-1}, u) \in E\}|\,. \quad (2)$$

During the I-th step, the meta-graph guided random walk first enumerates the count of edge types that meet the constraints among the edges originating from node v_{i-1}, and subsequently randomly picks an eligible edge type. Following this, it proceeds randomly to the next node through a chosen type of edge. If no fitting edge types are found, the random walk is terminated. This process generates a sequence of random walks $\mathcal{S}_{\mathcal{G}} = \{v_1, v_2, \cdots, v_L\}$, which has a length of L.

The GCN auto-encoder is designed to obtain topological feature embeddings, utilizing the excellent structure-capturing node feature fusion capability of GCN. To extract the features of each node, the random walk sequence $\mathcal{S}_{\mathcal{G}}$ is first realized as a subnetwork. This subnetwork is then transformed into a homogeneous network before being fed into the encoder for processing. The calculation of the GCN is as follows:

$$H^{(l+1)} = Relu(\widehat{D}^{-\frac{1}{2}} \widehat{A} \widehat{D}^{-\frac{1}{2}} H^{(1)} W^{(l)})\,, \quad (3)$$

where $H^{(l)}$ denotes the feature representation of nodes at the l-th layer. The modified subgraph adjacency matrix, $\widehat{A} = A + I_N$, includes self-connections, with I_N being the identity matrix. $\widehat{D}_{ii} = \sum_j \widehat{A}_{ij}$ is the degree matrix of $\widehat{A}$. The weight matrix, which can be trained, for layer l is represented as $W^{(l)}$. $Relu$ is a nonlinear activation function.

By stacking multiple layers, the GCN can capture more information. In this paper, a dual-scaler GCN is used as an encoder. The first layer generates the intermediate representation H_M, which is then used by the second layer to generate the final remote topology layer embedding H_T.

B. Semantic Aggregation Layer View

In the semantic aggregation layer, two attention mechanisms are implemented to evaluate the importance of different neighboring nodes and meta-paths. The required inputs are user meta-paths and item meta-paths. Meta-paths of user (item) are those that originate and terminate at nodes of the user (item) type.

First, every meta-path is represented as a reachable subgraph. The subgraph of the user (item) meta-path is exclusive to nodes of the identical type as the user (item), and these nodes will establish a connection if they are adjacent in terms of the meta-path. Next, using the GAT [17] layer, we obtain semantic information about each subgraph. The GAT layer possesses the ability to discern the significance of various meta-path neighbors and amalgamate information from them based on their weights. Ultimately, the consolidated embeddings are channeled into the succeeding attention layer.

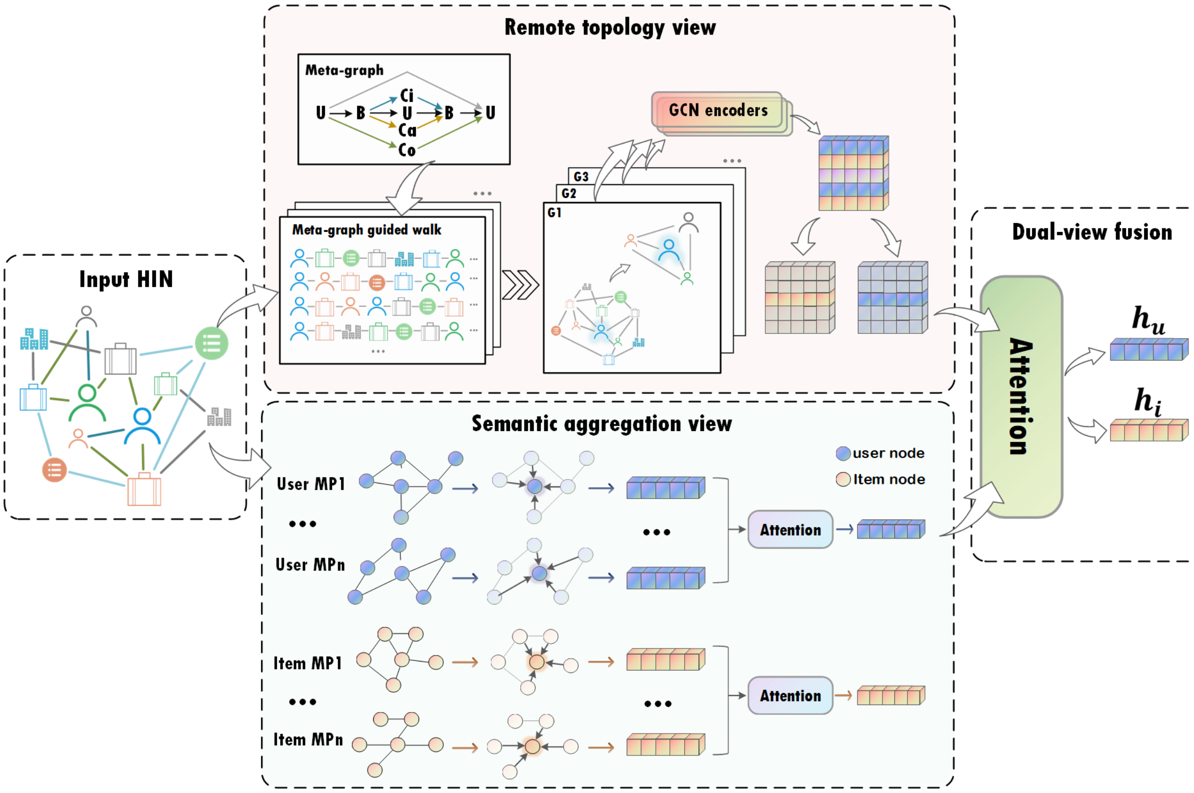


Fig. 2. Example of HIN for Douban movie dataset.

Given that distinct meta-paths in HIN do not share equal importance, after aggregating the information of nodes and edges on each meta-path, we also use an inter-meta-path aggregation layer to obtain the weights of different meta-paths and integrate them. Through an attention mechanism, this layer assigns weights to various meta-paths and integrates their embeddings, revealing a more comprehensive understanding of semantic information. Now, for node type A, suppose we have X meta-paths $\{\mathcal{P}_1, \mathcal{P}_2, \cdots, \mathcal{P}_X\}$. $\{H_1, H_2, \cdots, H_x\}$ are X sets of node embeddings after the GAT layer. Then, we use a Multilayer Perceptron (MLP) to transform the embeddings of nodes obtained from different meta-paths, in order to summarize each meta-path $P_i \in \mathcal{P}_X$:

$$s^{P_x} = \frac{1}{|\mathcal{V}_x|} \sum_{i \in \mathcal{V}_x} \sigma(MLP(h_i^{P_x})), h_i^{P_x} \in H_x \,, \tag{4}$$

where σ is the activation function and $\mathcal{V}_x$ is the set of nodes of the meta-path P_x.

Calculating the mean of all distinct node embeddings associated with meta-paths is as follows:

$$w^{P_x} = \vec{q}_{P_x}^{\,T} \cdot s^{P_x} \,, \tag{5}$$

$$\beta^{P_x} = \frac{\exp\left(w^{P_x}\right)}{\sum_{x=1}^{X} \exp\left(w^{P_x}\right)} \,, \tag{6}$$

where β^{P_x} denotes the normalized importance of the meta-path P_x, and $\vec{q}_{P_x}^{\,T}$ is the parameterized attention vector of the node type A.

Ultimately, by integrating the embeddings of every meta-path according to their respective weights, we derive the conclusive semantic embedding H_S for the nodes:

$$H_s = \sum_{x=1}^{X} \beta^{P_x} \cdot H_x \,. \tag{7}$$

C. Information Fusion and Recommendation

We have now obtained the embeddings H_T and H_S corresponding to the two sets of views, and we apply the Eqs. (4–6) again, which obtains the importance of each view through MLP and aggregates the embeddings of the two sets of views according to their respective weights to obtain the final node embedding H:

$$H = \beta^T \cdot H_T + \beta^S \cdot H_s \,, \tag{8}$$

where β^T and β^S are the two-view normalized importance calculated by Eqs. (4–6).

When users and items have their final embeddings, we compute their inner products to predict how well items and users match:

$$\hat{y}(u,i) = h_u^T h_i \,, \tag{9}$$

where h_u and h_i are the corresponding user and item embedding vectors for which the degree of matching needs to be predicted.

The recommendation method is subjected to training with BPR [31] loss, which is designed to elevate the scores of observed user-item interactions over those of unobserved ones,

$$\mathcal{L}_{rec} = \sum_{(u,i,j) \in O} -ln\sigma(\hat{y}(u,i) - \hat{y}(u,j)) \,, \tag{10}$$

where $O = \{(u,i,j) \mid (u,i) \in \mathbb{R}^{+}, (u,j) \in \mathbb{R}^{-}\}$ is representative of the training set, $\mathbb{R}^{+}$ signifies positive (interacting) user-item pairs, $\mathbb{R}^{-}$ signifies negative (non-interacting) user-item pairs, and $\sigma(\cdot)$ is indicative of the sigmoid function. In the training stage, the initial user and item node embeddings originate from those obtained via Matrix Factorization (MF) [31].

V. Experiment and Analysis

Experiments were carried out on two public datasets to validate of the DFHE and tackle these research inquiries:

- **RQ1:** What is the performance of DFHE in the recommendation task when juxtaposed with other baseline methods?
- **RQ2:** During the recommendation process, how important is each meta-path?
- **RQ3:** How's the efficiency of the method?

A. Experiment Settings

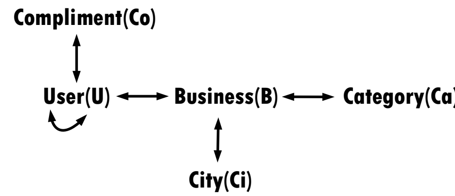
1) Datasets: To evaluate our method, we conducted experiments on two public datasets, Yelp and Douban movie. The Yelp dataset comprises 198,327 ratings of 14,284 businesses on a scale of 1-5 by 16,239 users, along with information on user social interactions and business attributes. The Douban movie dataset includes 13,367 score ratings of 12,677 movies on a scale of 1-5 by users. Additionally, the dataset also contains information on user social interactions, group affiliations, and movie attributes. Table I and Fig. 3 show the statistics and meta-schemas of the two datasets, respectively.

2) Experimental Methods and Metrics: To evaluate the performance of the method, we use a 'leave-one-out' evaluation method [30]. Items that the user interacted with are considered positive, while items that the user did not interact with are classified as negative. For each positive item, we arbitrarily chose 499 negative items and proceeded to rank the collective 500 items.

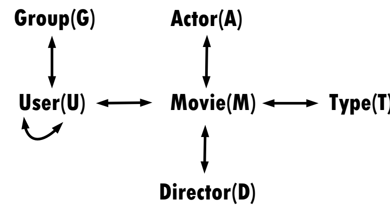
Two common metrics are employed in the evaluation of the proposed methods: *Hit Ratio at Rank k* (HR@k) and *Normalised Discounted Cumulative Gain at Rank k* (NDCG@k). These metrics are widely accepted in the research field and can accurately reflect the performance of the methods under consideration. HR@k accounts for whether test items are present in the top-k list, which is a direct measure of the recommendation accuracy. NDCG@k gauges the positioning of test-positive items within the ranking, thereby illuminating the efficacy of our method in terms of ranking. To assess the efficacy of top-K recommendation and preference ranking, we have adopted these two widely-utilised evaluation metrics.

3) Baselines: To validate the information extraction and recommendation effect of DFHE, we opted for both meta-path-based and non-meta-path-based recommendation methods for the purpose of comparison with our method.

BPR [31] is a sorting learning method that is commonly used in recommendation systems. The fundamental concept is to leverage users' implicit feedback data as a basis for recommending items to them.



(a) Yelp



(b) Douban Movie

Fig. 3. Meta-schema of two datasets; (a) is the meta-schema for the Yelp dataset, (b) is the meta-schema for Douban movie.

NCF [32] employs neural networks to simulate the interplay among users and items. This method improves the capture of user interests and preferences.

CKE [33] gleans semantic nuances of items by utilizing data including structure, text and visual elements from the knowledge base. This approach employs dual encoders for the derivation of embeddings.

CFKG [34] performs reasoning on a structured knowledge base and expands traditional collaborative filtering to learn heterogeneous knowledge base embedding.

GEMS [35] employs a concurrent genetic method to pinpoint relevant meta-structures for the purpose of recommendations. It dynamically assimilates information from these meta-structures through the use of a convolutional network.

KGAT [36] carries out the modeling of high-order relationships in a direct and comprehensive way, all within the structure of a graph neural network.

HERec [22] generates sequences of nodes by user and item meta-path-guided randomized wandering and design different fusion functions to comprehend the node embeddings.

McRec [37] utilizes a detailed meta-path-based context for top-N recommendations, requiring only a set of meta-paths that start from user-centric nodes and end with item-centric nodes.

RMS-HRec [26] applies Reinforcement Learning for Meta-Path Selection (RMS) to identify impactful meta-paths that might be missed in the manual creation phase.

TABLE I
STATISTICS OF THE DATASETS

Dataset (Density)	Node Relationship	Value A	Value B	Value (A-B)	Meta-path
	User (U) - Business (B)	16239	14284	198397	
Yelp	User (U) - User (U)	10580	10580	158590	UBCaBU/UBCiBU
(0.08%)	User (U) - Compliment (Co)	14411	11	76875	UBU/UCoU/BUB
	Business (B) - Category (Ca)	14180	511	40009	BCaB/BCiB/BUUB
	Business (B) - City (Ci)	14267	47	14267	
	User (U) - Movie (M)	13367	12677	1068278	
	User (U) - User (U)	2440	2294	4085	UMDMU/UMTMU
Douban Movie	User (U) - Group (G)	13337	2753	570047	UMU/UMAMU
(0.63%)	Movie (M) - Director (D)	10179	2449	11276	MAM/MUM
	Movie (M) - Actor (A)	11718	6311	33587	MTM/MDM
	Movie(M)-Type(T)	12678	38	27668	

4) Implementation Details: Pytorch and DGL [26] were used to implement all methods. The topological information extraction layer has a wander length of 100 and 20 wanders per node. For the recommendation part of DFHE, the embedding size is fixed at 64, the hidden layer size at 32, and the batch size at 50,000. The settings are such that the learning rate is established at 0.002, while the memory buffer size is configured to be 10,000. For the Yelp dataset, we set the maximum number of training rounds to 100. For Douban Movie, the maximum training rounds were set to 150. Additionally, here the maximum step size of HERec is set to 3, RMS-HRec to 4, and MCRec to 5.

The baseline recommenders were executed in standard configurations according to the code published in the original papers of the respective authors.

B. Experimental Results

1) Recommendation Performance (RQ1): Table II presents a comparative analysis of the comprehensive performance of DFHE across all datasets.

1) It's evident that DFHE exceeds the performance of other baseline methods in all three metrics on both datasets. This indicates that information extraction fusion under dual view is crucial, and our method efficiently utilizes the topological and semantic information in HIN for recommendation tasks.
2) Compared to MF-based methods (BPR, NCF) and GNN-based methods (GEMS, KGAT), DFHE's performance improvement is attributed to the meta-graphs and meta-paths concatenating a larger number of entity types and relationships during information extraction, which contain richer semantic information. Additionally, the effectiveness of modeling local graphs is also demonstrated.
3) When comparing with meta-path-based methods (HERce, McRec, RMS-HRec), the inclusion of remote topology view in DFHE allows for information extraction that is not entirely dependent on specific meta-paths, making the extraction of information more comprehensive. In addition, the comparison of RMS-HRec shows that the information in the network may be concentrated on a small number of "core meta-paths", and the increase of numerous meta-paths will result in an increase in the amount of computation, which also reflects the importance of view fusion as a means of complementing the information.

TABLE II
OVERALL PERFORMANCE COMPARISON

	Yelp			Douban Movie		
	HR@1	HR@3	NDCG@10	HR@1	HR@3	NDCG@10
BPR	0.0388	0.1025	0.1301	0.0529	0.1421	0.1768
NCF	0.0514	0.1251	0.1522	0.0622	0.1605	0.1974
CKE	0.0456	0.1092	0.1360	0.0612	0.1581	0.1947
CFKG	0.0572	0.1246	0.1477	0.0637	0.1650	0.2033
GEMS	0.0100	0.0294	0.0408	0.0262	0.0591	0.0740
KGAT	0.0415	0.1151	0.1388	0.0644	0.1669	0.2042
HERec	0.0389	0.0979	0.1215	0.0594	0.1613	0.1984
MCRec	0.0548	0.1317	0.1540	0.0928	0.1961	0.2236
RMS-HRec	0.0648	0.1484	0.1740	0.0997	0.2131	0.2400
DFHE	**0.0715**	**0.1558**	**0.1844**	**0.1083**	**0.2281**	**0.2546**

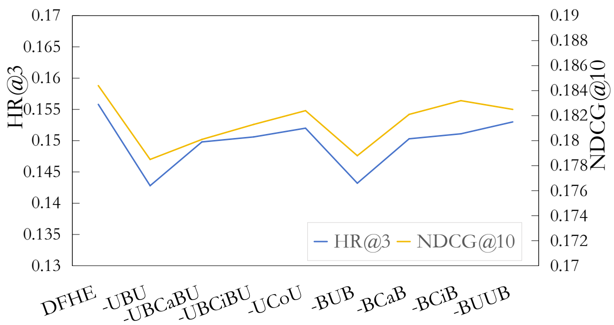
2) Meta-path Sensitivity Experiments (RQ2): Our method revealed a significant correlation between the selection of meta-paths and the final outcome of the recommendation. To study the sensitivity of each meta-path, we adopted the leave-one-out principle. Specifically, we adjusted certain meta-paths in the semantic extraction layer through addition or deletion. Then we evaluated how these alterations influenced the effectiveness of the method in comparison to its initial configuration. The meta-path sensitivity results are shown in Table III and Fig. 4 :

The results of Table III and Fig. 4 are analyzed below:

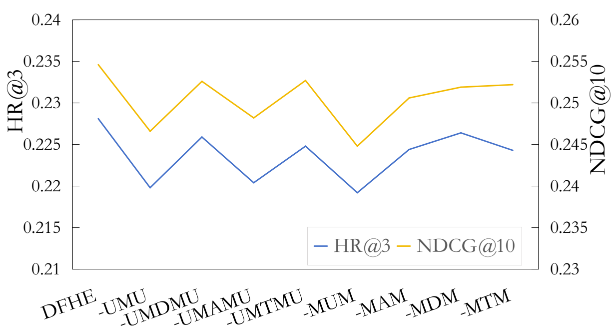
1) It is evident that the impact on final performance varies depending on which meta-path is removed from the set. It is observed that the meta-paths containing direct interaction information between users and projects encapsulate the core information within the meta-path set. Furthermore, the label information of users/items can serve as a complement to the user-item interaction information. Here, "user-item interaction information" includes the path information of users rating projects.

TABLE III
SENSITIVITY OF META-PATH ON DFHE

Yelp			Douban Movie		
Meta-path set	HR@3	NDCG@10	Meta-path set	HR@3	NDCG@10
DFHE	**0.1558**	**0.1844**	**DFHE**	**0.2281**	**0.2546**
-UBU	0.1428	0.1785	-UMU	0.2198	0.2466
-UBCaBU	0.1498	0.1801	-UMDMU	0.2259	0.2526
-UBCiBU	0.1506	0.1813	-UMAMU	0.2204	0.2482
-UCoU	0.1520	0.1824	-UMTMU	0.2248	0.2527
-BUB	0.1432	0.1788	-MUM	0.2192	0.2448
-BCaB	0.1503	0.1821	-MAM	0.2244	0.2506
-BCiB	0.1511	0.1832	-MDM	0.2264	0.2519
-BUUB	0.1530	0.1825	-MTM	0.2243	0.2522



(a) Yelp



(b) Douban Movie

Fig. 4. Meta-path sensitivity.

2) Despite the loss of individual path information, the final results do not significantly decline compared to the complete path set in Table I. This stability is attributed to the dual-view fusion, which prevents significant performance degradation even in the absence of some effective meta-paths.

3) Method Efficiency Comparison (RQ3): Fig. 5 displays the training efficiency of DFHE and all baseline recommendation methods.

Our method necessitates the encoding of subgraphs generated from walk sequences under the guidance of a meta-graph, as well as the extraction of semantic context from multi-round meta-paths, which incurs a certain time cost. However, we found that our method is faster compared to other GNN-based methods (GEMS [35], KGAT [36]) and reinforcement learning-based methods (RMS-HRec [26]), and it is not significantly slower than other methods. Moreover, the performance of our method has improved. We believe that this effectively balances performance and efficiency.

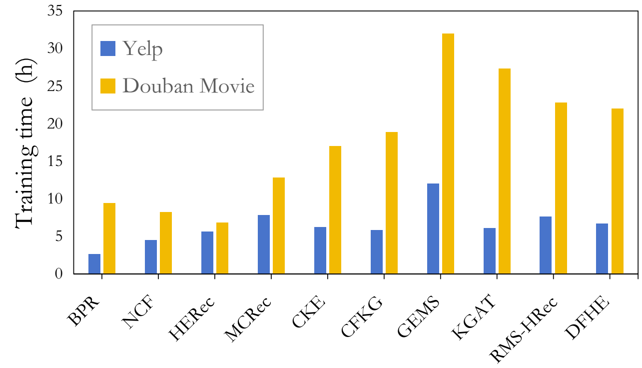


Fig. 5. Training efficiency comparison.

VI. CONCLUSION AND FUTURE WORK

In this paper, a HIN embedding recommendation method (DFHE) utilizing dual-view fusion is proposed. This method seizes both the distant topological data and node context semantic details in HIN. It employs a multi-layer attention mechanism to amalgamate the remote topology view embeddings and the semantic extraction view embeddings, resulting in the final embedding. It focuses on information between remote nodes, within and between meta-paths, and the subgraphs generated by remote node wandering sequences compensate to some extent for the problem of singular information on a given meta-path, which is the key to improve information capture. DFHE has executed recommendation operations on two datasets, namely Yelp and Douban Movie. It further scrutinized the sensitivity of meta-paths and the efficiency of training. The encouraging outcomes from the experiments attest to the efficacy and logic of the proposed DFHE.

As a result of efficiency considerations and selection of representative meta-paths, automatic sequence semantic discovery also becomes a challenging task in the future. At the same time, since real-world user preferences change over time, how to cope with dynamic HINs for recommendation is also worth investigating.

REFERENCES

[1] D. Liu and L. Li, “A node clustering algorithm for heterogeneous information networks based on node embeddings”, *Multimedia Tools and Applications*, vol. 83, pp. 3745-3766, 2024. DOI: 10.1007/s11042-023-15245-9

[2] Y. Chen, X. Xie, and W. Weng, “Content and structure based attention for graph node classification”, *Journal of Intelligent & Fuzzy Systems*, pp. 1–15, 2024. DOI:10.3233/JIFS-223304

[3] M. Han, H. Zhang, W. Li, and Y. Yin, “Semantic-guided graph neural network for heterogeneous graph embedding”, *Expert Systems with Applications*, vol. 232, pp. 120810-120820, 2023. DOI: 10.1016/j.eswa.2023.120810

[4] S. Zhang, J. Zhang, X. Song, S. Adeshina, D.Zheng, C. Faloutsos, and Y. Sun, “PaGE-Link: path-based graph neural network explanation for heterogeneous link prediction”, in *Proceedings of the ACM Web Conference*, pp. 3784–3793, 2023. DOI: 10.1145/3543507.3583511

[5] Z. Zhang, H. Xu, Y. Li, Z. Zhai, and Y. Ding, “NAGNE: node-to-attribute generation network embedding for heterogeneous network”, *Applied Sciences*, vol. 14, no. 3, pp. 1053-1065, 2024. DOI: 10.3390/app14031053

[6] Z. La, Y. Qian, H. Leng, T. Gu, W.Gong, and J. Chen, “MC-GAT: multi-channel graph attention networks for capturing diverse information in complex graphs”, *Cognitive Computation*, vol. 16, no. 2, pp. 595-607, 2024. DOI: 10.1007/s12559-023-10222-8

[7] B. Zeng, J. Chi, P. Hong, G. Lu, D. Zhang, and B. Chen, “Context-aware graph embedding with gate and attention for session-based recommendation”, *Neurocomputing*, vol. 574, pp. 127221-127235, 2024 DOI: 10.1016/j.neucom.2023.127221

[8] 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*, pp. 701–710, 2014. DOI: 10.1145/2623330.2623732

[9] J. Tang, M. Qu, M. Wang, M. Zhang, J. Yan, and Q. Mei, “Line: large-scale information network embedding”, in *Proceedings of the 24th International Conference on World Wide Web*, pp. 1067–1077, 2015. DOI: 10.1145/2736277.2741093

[10] A. Grover and J. Leskovec, “Node2vec: scalable feature learning for networks”, in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 855–864, 2016. DOI:10.1145/2939672.2939754

[11] T. Liu, J. Yin, and Q. Qin, “MFHE: multi-view fusion-based heterogeneous information network embedding”, *Applied Sciences*, vol. 12, no.16, pp. 8218-8232, 2022. DOI:10.3390/app12168218

[12] Y. Dong, N. V. Chawla, and A. Swami, “Metapath2vec: scalable representation learning for heterogeneous networks”, in *Proceedings of the 23rd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 135–144, 2017. DOI: 10.1145/3097983.3098036

[13] Y. He, Y. Zhang, L. Qi, D. Yan, and Q. He, “Outer product enhanced heterogeneous information network embedding for recommendation”, *Expert Systems with Applications*, vol. 169, pp. 114359, 2021. DOI: 10.1016/j.eswa.2020.114359

[14] Z. Zhao, Zhang X, H. Zhou, C. Li, and M. Gong, “HetNERec: heterogeneous network embedding based recommendation”, *Knowledge-Based Systems*, vol.204, pp. 106218–106228, 2020. DOI: 10.1016/j.knosys.2020.106218

[15] T. Mikolov, I. Sutskever, K. Chen, G. S. Corrado, and J. Dean, “Distributed representations of words and phrases and their compositionality”, *Advances in neural information processing systems*, vol. 26, 2013. DOI: 10.7551/mitpress/7503.001.0001

[16] D. Zhang, J. Yin, X. Zhu, and C. Zhang, “Metagraph2vec: Complex semantic path augmented heterogeneous network embedding”, in *Proceedings of the Pacific-Asia Conference on Knowledge Discovery and Data Mining*, pp. 196–208, 2018. DOI: 10.1007/978-3-319-93037-4_16

[17] P. Velickovic, G. Cucurull, A. Casanova, A. Romero, P. Lio, and Y. Bengio, “Graph attention networks”, *arXiv preprint*, arXiv:1710.10903, 2017. DOI: 10.48550/arXiv.1710.10903

[18] X. Wang, H. Ji, C. Shi, B. Wang, Y. Ye, P. Cui, and P. Yu, “Heterogeneous graph attention network”, in *The World Wide Web Conference*, pp. 2022–2032, 2019. DOI: 10.1145/3308558.3313562

[19] X. Fu, J. Zhang, Z. Meng, and I. King, “Magnn: metapath aggregated graph neural network for heterogeneous graph embedding”, in *Proceedings of the Web Conference*, pp. 2331–2341, 2020. DOI: 10.1145/3366423.3380297

[20] H. Du, C. Ma, D. Lu, and J. Liu, “HHSE: heterogeneous graph neural network via higher-order semantic enhancement”, pp. 1–23, 2024. DOI: 10.1007/s00607-023-01246-x

[21] J. Yu and X. Li, “Heterogeneous graph contrastive learning with meta-path contexts and weighted negative samples”, in *Proceedings of the 2023 SIAM International Conference on Data Mining (SDM)*, pp. 37–45, 2023. DOI: 10.1137/1.9781611977653.ch5

[22] C. Shi, B. Hu, W. Zhao, and P.Yu, “Heterogeneous information network embedding for recommendation”, *IEEE Transactions on Knowledge and Data Engineering*, vol. 31, no.2, pp. 357-370, 2018. DOI: 10.1109/TKDE.2018.2833443

[23] T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Efficient estimation of word representations in vector space”, *arXiv preprint*, arXiv:1301.3781, 2013. DOI: 10.48550/arXiv.1301.3781

[24] D. Wang, P. Cui, and W. Zhu, “Structural deep network embedding”, in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 1225–1234, 2016. DOI: 10.1145/2939672.2939753

[25] S. Forouzandeh, K. Berahmand, R. Sheikhpour, and Y. Li, “A new method for recommendation based on embedding spectral clustering in heterogeneous networks (RESCHet)”, *Expert Systems with Applications*, pp. 120699–120700, 2023. DOI: 10.1016/j.eswa.2023.120699

[26] W. Ning, R. Cheng, J. Shen, N. H. Haldar, B. Kao, X. Yan, N. Huo, W. K. Lam, T. Li, and B. Tang, “Automatic meta-path discovery for effective graph-based recommendation”, in *Proceedings of the 31st ACM International Conference Information and Knowledge Management*, pp. 1563–1572, 2022. DOI: 10.1145/3511808.3557244

[27] H. Wang, J. Wang, J. Wang, M. Zhao, W. Zhang, F. Zhang, X. Xie, and M. Guo, “GraphGAN: graph representation learning with generative adversarial nets”, in *Proceedings of the AAAI Conference on Artificial Intelligence*, vol.32, no.1, 2018. DOI: 10.1609/aaai.v32i1.11872

[28] T. N. Kipf and M. Welling, “Semi-supervised classification with graph convolutional networks”, *arXiv preprint*, arXiv:1609.02907, 2016. DOI: 10.48550/arXiv.1609.02907

[29] D. Jin, C. Huo, C. Liang, and L. Yang, “Heterogeneous graph neural network via attribute completion”, in *Proceedings of the Web Conference*, pp. 391–400, 2021. DOI: 10.1145/3442381.3449914

[30] Z. Han, M. U. Anwaar, S. Arumugaswamy, T. Weber, T. Qiu, H. Shen, Y. Liu, and M. Kleinsteuber, “Metapath-and entity-aware graph neural network for recommendation”, *arXiv preprint*, arXiv:2010.11793, 2020. DOI:10.48550/arXiv.2010.11793

[31] S. Rendle, C. Freudenthaler, Z. Gantner, and L. Schmidt-Thieme, “BPR: bayesian personalized ranking from implicit feedback”, in *Proceedings of the 25th Conference on Uncertainty in Artificial Intelligence*, pp. 452–461, 2009. DOI: 10.48550/arXiv.1205.2618

[32] 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*, pp. 173–182, 2017. DOI: 10.1145/3038912.3052569

[33] F. Zhang, N. J. Yuan, D. Lian, X. Xie, and W. Ma. “Collaborative knowledge base embedding for recommender systems”, in *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 353–362, 2016. DOI: 10.1145/2939672.2939673

[34] Y. Zhang, Q. Ai, X. Chen, and P. Wang, “Learning over knowledge-base embeddings for recommendation”, *arXiv preprint*, arXiv:1803.06540, 2018. DOI: 10.48550/arXiv.1803.06540

[35] Z. Han, F. Xu, J. Shi, Y. Shang, H. Ma, P. Hui, and Y. Li, “Genetic meta-structure search for recommendation on heterogeneous information network”, in *Proceedings of the 29th ACM International Conference on Information and Knowledge Management*, pp. 455–464, 2020. DOI: 10.1145/3340531.3412015

[36] X. Wang, X. He, Y. Cao, M. Liu, and T.-S. Chua, “Kgat: Knowledge graph attention network for recommendation”, in *Proceedings of the 25th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, pp. 950–958, 2019. DOI: 10.1145/3292500.3330989

[37] 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 and Data Mining*, pp. 1531–1540, 2018. DOI: 10.1145/3219819.3219965

Jinlong Ma received his Doctor of Engineering degree in Information and Communication Engineering from Harbin Institute of Technology. He currently serves as an Associate Professor at the School of Information Science and Engineering at Hebei University of Science and Technology. His research interests encompass the dynamics of complex networks and network representation learning.

Runfeng Wang is currently pursuing his Master’s degree at the School of Information Science and Engineering at Hebei University of Science and Technology. His areas of interest include big data in networks and network embedding.