Computing and Informatics, Vol. 45, 2026, 1129–1155, doi: 10.31577/cai_2026_5_1129

ENHANCING ONLINE LEARNING COMMUNITY DISCOVERY THROUGH GRAPH NEURAL NETWORKS AND DATA OPTIMIZATION

Bo Zhang

College of Information, Mechanical and Electrical Engineering
Shanghai Normal University, Shanghai, 200234, China
&
Shanghai Engineering Research Center of Intelligent Education and Bigdata
Shanghai Normal University, Shanghai, 200234, China
&
Shanghai University of Electric Power, Shanghai, 201306, China
e-mail: zhangbo@shnu.edu.cn

Di Cai, Xinyi Zhang

College of Information, Mechanical and Electrical Engineering
Shanghai Normal University, Shanghai, 200234, China
e-mail: 1000526998@smail.shnu.edu.cn, ndaynday@163.com

Hao Wang

Shanghai Newtouch Software Co., Ltd., Shanghai, 200127, China
e-mail: hao.wang@newtouch.com

Zhonghua Zheng

Anhui Boryou Information Technology Co., Ltd., Hefei, 230026, China
e-mail: zzh@boryou.com

Meizi Li*, Ru Yang*

College of Information, Mechanical and Electrical Engineering
Shanghai Normal University, Shanghai, 200234, China
e-mail: {limeizi, yangru}@shnu.edu.cn

Abstract. As digital platforms continue to reshape the way knowledge is acquired and shared, there is a critical need for effective community discovery algorithms that can operate in sparse data environments. Traditional methods often fail to capture the complex dynamics of online educational platforms, where explicit relationships among learners are typically underrepresented. Therefore, to address the challenges of sparse inter-learner connections within online learning communities, this paper proposes a novel framework that introduces the "Cyber Learning Community" model which reconceptualizes learner relationships by emphasizing collaborative and supportive interactions. Specifically, it combines a modified graph neural network (GNN) architecture and a bespoke community discovery algorithm to unearth and reinforce the latent networks within large datasets. The proposed methodology involves two primary tasks: predicting learners' interest labels from their online behavior, and identifying robust community structures through the innovative use of the Discrete Beetle Antennae Search algorithm (DBAS). The effectiveness of our approach is demonstrated through extensive experiments on real-world datasets, showing marked improvements in community coherence and learning engagement. The results indicate that our methods not only enhance the detection and formation of educational communities but also suggest scalable applications for broader social network analysis.

Keywords: Online learning, community detection, GNN

Mathematics Subject Classification 2010: 68T05, 68R10

1 INTRODUCTION

With the development of network technology, the channels for learners to acquire knowledge have expanded, and traditional learning methods are gradually being replaced by informalized education due to the constraints of time, space, and other factors [1]. In recent years, online education platforms have broken the time and space constraints of traditional classrooms. Increasingly, social platforms prioritize the creation and sharing of online knowledge resources. Active users on these platforms contribute to knowledge construction through their output and accelerate knowledge dissemination through their social interactions [2, 3]. It is obvious that the behavior of learners in online knowledge sharing platforms (K-S platform) can have a great impact on both the platform and other learners [4]. Now online learners are increasingly and spontaneously gathering and forming online learning communities. From a sociological perspective, an online learning community serves as a public space where participants engage in group learning behaviors. All interactions and learning activities occur within the online environment. Some experts argue that the structure of these communities can profoundly influence learners'

* Corresponding authors

educational outcomes [5, 6]. Garrison et al. [7] believe that deep and meaningful network learning takes place in a community. Thus, the discovery of network learning communities plays an important role in the subsequent research and analysis of group learning behavior in complex networks, and further explores the various laws that exist in network learning communities.

A significant issue with current online learning communities is the scarcity of explicit relationships between learners. This lack of connectivity undermines the community structure, which ideally should feature strong internal relationships and sparse external connections. Traditional community discovery algorithms frequently encounter challenges in effectively addressing this sparse data scenario [8]. These algorithms, such as modularity-based algorithms [9] and the Girvan-Newman algorithm [10], are typically designed to identify densely connected groups of nodes rather than deal with sparse explicit relationships between learners. Therefore, a new approach is needed that can be adapted to the unique structure of online learning communities.

On the other hand, the openness of online K-S platforms has led to a significant increase in the amount of data. While the big data environment provides favorable conditions for the application of deep learning methods, especially in information processing and analysis, it also poses challenges. The huge amount of data makes it difficult to mine it effectively, which directly affects the measurability of network structure. Most studies on community structure mining [11, 12, 13, 14] have focused on describing the observable community structure, ignoring the challenges posed by this unpredictability.

Furthermore, while deep learning methods, particularly graph neural network (GNN) models, have made significant strides in learning user feature representations, challenges such as neighborhood construction difficulty and community labeling bias introduced by supervised/semi-supervised learning methods [15, 16, 17, 18] during model training have constrained their effectiveness. For example, Hamilton et al. [15] note in their GraphSAGE work that although GNNs can effectively learn node features, neighborhood sampling and labeling bias remain challenges. These technical challenges highlight the shortcomings of existing approaches when dealing with online learning community data.

In view of the above problems, this paper proposes a new approach called "Cyber Learning Community". This method is derived from the concept of an online learning community, but the relationship between learners is constructed in a different way. In the Networked Learning Companion Community, the relationship between learners is called a Learning Companion Relationship, which is different from the traditional social relationship and focuses more on collaboration and mutual support among learners. Within this situation, this approach aims to solve the problem that cannot be solved by existing methods, i.e., how to build closer relationships among learners while dealing with community discovery and structure mining in a big data environment. The contributions of this paper are as follows.

1. It obtains the effective characteristics representation of the massive data learners have in the online K-S platform. This not only further analyzes the learning behavior habits and learning interest preferences of learners, but also provides a feasible solution for mining the data of the online K-S platform;
2. It proposes a community discovery methodology to deal with community discovery on heterogeneous information networks which focuses on the design of sparse network structure complementation and efficient community structure detection models;
3. It designs an experimental research program for online K-S platforms that can migrate advanced algorithms to application scenarios while retaining their educational characteristics and provides more research ideas and relevant realistic basis for solving online educational problems.

The subsequent sections of this paper are structured as follows: Section 2 provides an overview of prior research, some new concepts and definitions to describe the proposed model are given in Section 3, followed by a detailed explanation of our methodology in Section 4, which encompasses the framework of our interest label prediction model and the community discovery algorithm. Then, Section 5 presents thorough empirical evaluations, including experiments designed to evaluate the efficacy of our prediction model and to identify genuine social communities. The last section, Section 6, summarizes our contributions and delves into potential avenues for future research.

2 RELATED WORK

Community discovery is a central research problem in network science, involving the identification of densely connected groups of nodes in a network. These community structures are prevalent in many types of networks, such as social networks, protein interaction networks, and citation networks, and are important for understanding the structure and function of networks.

2.1 Traditional Community Detection

Early community discovery studies relied heavily on graph theory and physics-based approaches. Modularity optimization [19] was proposed as one of the first basic methods to identify community structure by comparing the actual edge density in the network with the randomly expected edge density. Since then, various algorithms based on modularity optimization have been proposed. Among these, Louvain's algorithm [20] has garnered significant attention due to its efficient handling of large networks. However, despite improvements in computational efficiency, the stability and consistency of its results are susceptible to initial conditions and may vary across different runs.

Spectral clustering methods use the eigenvectors of the Laplace matrix of a network for community segmentation [21]. While this method excels in handling the

global structure of networks, it demands substantial computational resources, particularly when applied to large networks, potentially making its computational cost prohibitive. On the other hand, the hierarchical clustering method [22] constructs a hierarchical structure of communities by gradually merging or splitting nodes, which is suitable for discovering nested community structures.

Information flow methods such as Infomap [23], which identifies communities by minimizing the amount of information describing random wandering paths, are particularly effective at revealing patterns of information flow in networks. In addition, density-based approaches [24] and the label propagation-based approach [25], which focus on local variations in the density of nodes in a network and the identification of communities through the propagation of node labels, respectively, have also been applied to community discovery.

While traditional community discovery methods provide a basic framework for understanding and mining community structure in networks in the early stages of network science, they face several limitations. These methods tend to rely on predefined graph-theoretic metrics, such as modularity, which can lead to resolution limitations and make it difficult to detect smaller communities or deal with large networks with complex topologies. In addition, these methods suffer from shortcomings in computational efficiency, stability, and consistency, especially when the amount of network data to be processed is large and dynamically changing. While spectral clustering and hierarchical clustering methods offer different perspectives on community detection, they still face challenges in terms of computational complexity, optimal parameter selection, and interpretability.

2.2 GNN-Based Community Detection

In recent years, with the rapid development of graph neural networks (GNNs), community discovery methods based on GNNs have become a hot research topic. Graph Convolutional Network (GCN) [26] provides a new perspective on community discovery by applying convolutional operations to graph structures to learn the feature representation of nodes. Graph Auto-Encoder (GAE) and Variational Graph Auto-Encoder (VGAE) [27] learn node embeddings by reconstructing the adjacency matrix of a graph, which in turn can be used for community delineation. However, the performance of these methods may degrade when dealing with networks with strong community overlap or dynamics.

Graph Attention Network (GAT) [28] introduces an attention mechanism that allows the model to dynamically focus on important neighboring nodes, thus improving the accuracy of community discovery. In addition, some studies [29] have focused on combining traditional community discovery methods with GNNs to take advantage of both. For example, CommunityGNN [30] achieves more accurate community delineation by fusing the node features learned by GNNs with the advantages of traditional community discovery algorithms. However, this fusion method needs to be carefully designed to effectively combine the advantages of both methods while avoiding adding too much computational complexity.

While GNN-based community discovery methods offer unprecedented capabilities in handling node characteristics and network structure, they encounter several limitations. First, these methods may suffer performance degradation when confronted with networks exhibiting high community overlap or temporal dynamics, as standard GNN architectures may struggle to capture such complexities effectively. Second, the training and tuning processes of GNN models are computationally intensive and resource-consuming, restricting their application to large-scale networks. Moreover, the interpretability of GNNs remains a challenge, particularly in specific application scenarios where understanding how models recognize communities is crucial.

In summary, current graph representation learning methods often overlook the local graph structure and intricate node relationships. Traditional community discovery algorithms are inefficient for large-scale graphs, while deep learning-based approaches can handle large-scale data but are influenced by factors like network structure, node attributes, and edge density, affecting their effectiveness. Hence, there remains a need for effective methods to address the increasingly complex task of online community detection.

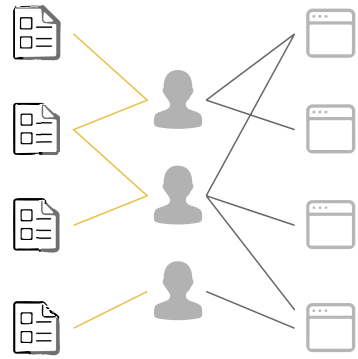
3 PROBLEM STATEMENT

To better understand the research task of this paper, the symbols and related definitions used in this paper are explained in Table 1, and the relationships between the learning behavior heterogeneous graph and the nodes based on the learning behavior design in this paper are shown in Figure 1. In Figure 1 a), learners learn different types of learning resources by different learning behaviors. And different learners who are connected by the same learning resource will become learning partners, as shown in Figure 1 c).

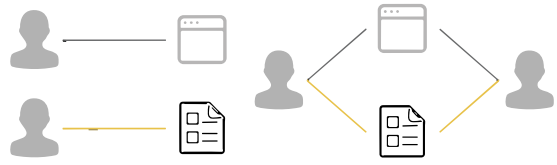
Notations	Definitions
G_K	Original heterogeneous graph
G_P	Learning companionship relationship network
G_B	Metapath-based bipartite graph
$C = \{C_1, C_2, \ldots, C_m\}$	Set of learning partner communities
$W = \{W_1, W_2, \ldots, W_n\}$	Set of learning resources
$x = \{(W_i \mid i \in [0, n])\}$	Learning behavior sequence
u/v	Learner/Resource node
η_i	Structure/Similarity threshold
ϵ	Existence determination threshold

Table 1. Summary of notations

In order to more accurately describe the model proposed in this paper, some new concepts are given below.



a) Online knowledge network



b) Learning behavior c) Learning partner

Figure 1. Online K-S network and relationships

Definition 1 (Heterogeneous Graphs and Meta-paths)**.** A graph consisting of more than one type of nodes and edges is called a heterogeneous graph [31]. The original heterogeneous graph mined from learning behavior data is denoted as $G_k = (V, E)$, where V contains learners and learning resources, and E denotes the learning behaviors. A sequence of different nodes in the heterogeneous graph is called a meta-path, which is represented by the formalization: $V_1 \xrightarrow{E_1} V_2 \xrightarrow{E_2} \cdots \xrightarrow{E_l} V_{l+1}$ and implies a set of complex semantic information of learning behaviors.

Definition 2 (Learning companion network.)**.** $G_p = (V, E)$ denotes the learning companion relationship network which is different from the original heterogeneous graph and keeps only one kind of nodes, and the edges indicate learning partner relationships. In the process of learning partner relationship mining, function f represents the operation of taking the intersection of the set of learning resource neighbors of a learner node and it is considered that there is a learning partner relationship exists between learners u_i, u_j if $f(u_i, u_j) \geq \epsilon$.

Definition 3 (Networked community of learners.)**.** The topology of community C_i satisfies $S_{in} = \frac{L(C_i, C_i)}{|C_i|} > \eta_1$, $S_{out} = \frac{L(C_i, \bar{C}_i)}{|C_i|} < \eta_2$, where η_1 and η_2 denote the intra and inter connectivity thresholds, respectively. Besides, $L(C_i, C_j) = \sum_{u_p \in C_i, u_q \in C_j} a_{pq}$ denotes the connectivity between communities, where a_{pq} denotes the node u_i, u_j

weights of connected edges between nodes. In addition, the learning companion relationship needs to satisfy $S_{la} = \frac{\sum y_{u_p}^T \cdot y_{u_q}}{\|y_{u_p}\|_2^2 \cdot \|y_{u_q}\|_2^2} < \eta_3$, which η_3 denotes the learning behavior similarity threshold.

4 METHODOLOGY

4.1 Overall

In this section, we propose a novel community discovery algorithm based on interest labels, which can be divided into two main parts: interest label extraction and community structure detection. The overall modeling framework is shown in Figure 2. Initially, the feature representation of learners and learning resources is acquired through a heterogeneous graph within an attention network. Subsequently, a multi-layer perceptron is employed to predict learners' interest labels based on this representation. Then, a bipartite graph representing learning behaviors via metapaths is integrated to form the final learning partner network. Using the DBAS algorithm proposed in this paper, this network is segmented to delineate the community structure of learning partners as intended. Finally, communities of learners maintain close collaborative ties, sharing similar learning interests and engaging in interest-driven cooperative learning within their respective communities.

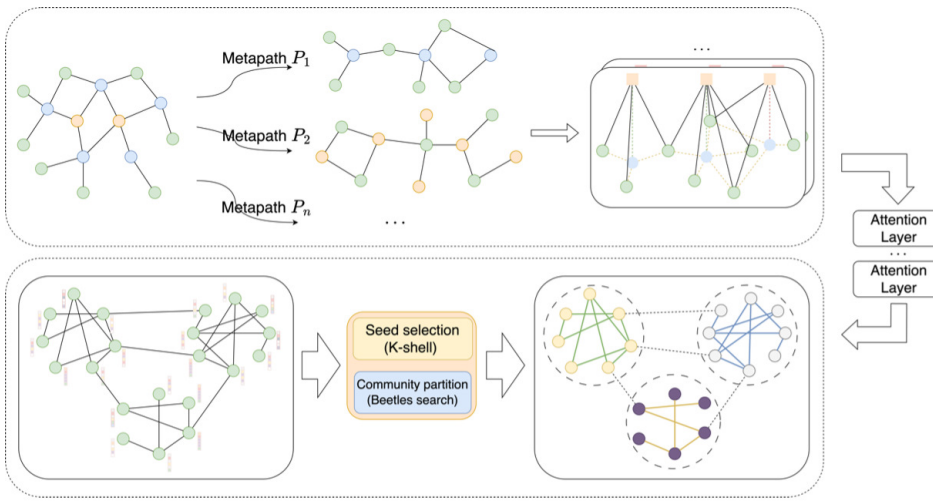


Figure 2. Overall modeling framework

4.2 Learning Behavior Interest Label Prediction

In a real online learning process, learners generate sequences of learning behaviors that are used to record the learner's learning process. This part of the work aims to mine the learner's learning interests using the learner's learning behavior sequence, while reducing the computational complexity of the subsequent work. To address the problem of mining learning behavior features, this paper proposes a label prediction model based on graph neural network architecture, which includes the following two aspects.

4.2.1 Learning Behavior Heterograph Construction

In this module, the correspondence between learners and learning resources in a sequence of learning behaviors is used to construct a learning behavior heterograph. The learning behaviors of all learners over a specific period are denoted as X. Each learner u_i's learning behavior sequence is represented as $x_{u_i} = \{(w_r, w_s, \ldots, w_t) \mid r, s, \ldots, t \in [1, n]\}$, where w_i denotes the learning resources comprising the sequence. The set of neighbor nodes N_{u_i} in the learning behavior heterogeneous graph consists of all learning resources corresponding to the learner's behaviors. Different types of learning behaviors correspond to different types of edges connecting learners and resources in the graph G_K. Specifically, any two learners u_i, u_j may share a 2J-hop neighbor relationship, where $J \in R$. The online learning behaviors of learners are influenced not only by their individual actions but also by social relationships during the learning process. To construct the feature vector for a learner node in the heterogeneous graph, the sequence of learning behaviors and social attribute coding are concatenated. The initial feature vector is denoted as $z^0_{u_i} = (z^0_1, z^0_2, \ldots, z^0_g)$. Similarly, the features of learner incorporate both the sequence of learning behaviors and the learner's social attribute, forming the initial feature vector $r_{w_i} = (r_1, r_2, \ldots, r_h)$.

4.2.2 Learning Behavior Feature Mining

To uncover implicit features in the learning behavior heterogeneous graph, this study enhances the feature embedding method using random walk techniques, thereby maximizing the utility of the constructed graph. Initially, the heterogeneous graph G_k is decomposed based on meta-paths to generate subgraphs corresponding to these paths. Next, neighborhood sampling and feature fusion are conducted on target learner nodes across various subgraphs of the learning behavior heterogeneous graph. Finally, feature vectors from different subgraphs are concatenated to derive the learning behavior features of the learner nodes, denoted as $z_{u_i} = (z_1, z_2, \ldots, z_g)$. The specific steps are as follows:

Step 1: Sequence Alignment. This step aims to construct the domain structure of the target learner node. For the learner u_i and the neighbor u_j which is connected through metapath with the maximum number of hops, the sequences of learning behaviors are x_{u_i}, x_{u_j}. The alignment comparison process starts from

the first position of these two sequences. A score is given for each position of the aligned sequence, and the total score is the sum of the scores for each position. In the alignment operation, there are D times insertion operations, where each operation has a penalty ω_1 and I times deletion operations, where each operation gains penalty ω_2. The loss in the whole alignment process is shown as L_{sa}. Then the sequence similarity $sim(u_i, u_j)$ of learning behavior between learners can be obtained as follows:

$$L_{sa} = \omega_1 \cdot D + \omega_2 \cdot I, \tag{1}$$

$$\mathrm{sim}(u_i, u_j) = 1 - \frac{L_{sa}}{|x_{u_i}| + |x_{u_j}|}. \tag{2}$$

In the actual online learning process, learners are more likely to be influenced by the learner with higher similarity of learning behaviors, so it is assumed here that the sequence similarity of learning behaviors reaches a certain threshold value μ when the two learners can have to cause influence between them, i.e.:

$$A_{M_p}(u_i, u_j) = \begin{cases} 1, & \mathrm{sim}(u_i, u_j) \geq \mu \wedge u_j \in N_{M_p}(u_i), \\ 0, & \text{otherwise}, \end{cases} \tag{3}$$

where A_{M_p} is the corresponding adjacency matrix of network G_k under the metapath M_p, and $N_{M_p}(u_i)$ is the set of neighboring nodes connected by learner node u_i through the metapath M_p.

Step 2: Interest Feature Extraction. Firstly, feature fusion of meta-path information is needed. For the instance of metapath M_p which is connecting two nodes u_i, u_j, it is denoted as $p(u_i, u_j)$ and the set of nodes is denoted as $S_p(u_i, u_j)$. The fused features of the metapath information $h_p(u_i, u_j)$ is formulated as:

$$h_{p(u_i,u_j)} = \frac{\sum_{v \in S_p(u_i,u_j)} h_v}{|S_{p(u_i,u_j)}|}, \tag{4}$$

where h_v is the feature vector of the initial features of the node after linear transformation to ensure that the feature dimensions of different types of nodes are the same. After that, the meta-path instances are fused with weights by applying an attention mechanism to the metapaths connected by the same node:

$$e^p_{u_i u_j} = \mathrm{LeakyReLU}(\alpha_p^T \cdot [h_{u_i} \| h_{p(u_i,u_j)}]), \tag{5}$$

$$\alpha^p_{u_i u_j} = \frac{\exp(e^p_{u_i u_j})}{\sum_{u_j \in N_{M_p}(u_i)} \exp(e^p_{u_i u_j})}, \tag{6}$$

$$z^{M_p}_{u_i} = \sigma\left(\sum_{u_j \in N_{M_p}(u_i)} \alpha^p_{u_i u_j} \cdot h_{p(u_i,u_j)} \right), \tag{7}$$

where α_p denotes the metapath M_p of the attention parameter. Finally, the learner u_i's feature vectors on all subgraphs are spliced to obtain the final feature vector representation $z_{u_i} = concat(z_{u_i}^{M_1}, \ldots, z_{u_i}^{M_p})$.

Finally, the learning behavioral features Z are fed as the input of MLP to predict the learner's interest labels Y. And the cross-entropy loss function is used as the loss function for prediction $Loss = -\sum_{i=1}^{|U|} y_{u_i} log\hat{y_{u_i}}$, where $|U|$ denotes the number of learners in the learning behavior heterogeneous graph, and y_{u_i} denotes the learner u_i's actual interest labels, and $\hat{y_{u_i}}$ denotes the interest label output by MLP.

4.3 Learning Companion Community Discovery

After obtaining the learners' interest labels, the main work in this part is to mine the potential community structure among learners based on the interest characteristics, and help the learners better carry out cooperative learning and improve the group distribution of learning effects. First, based on the above research, the learning behavior heterogeneous graph is de-heterogenized to obtain the learning partner relationship network containing only learner nodes to facilitate the use of the community discovery algorithm. Subsequently, the improved community discovery algorithm is used in the learning partner relationship network to excavate the community of networked learners therein.

4.3.1 Learning Companionship Mining

Learners generate learning companionships in online companion communities through learning resources that serve as a medium, mapping the fact that learners with similar learning habits are more suitable to be matched together for cooperative learning in the real learning process. Therefore, we use the method of extracting bipartite graphs to mine learning partner relationships. Starting with the heterogeneous graphs in Section 3, we first identify and extract relevant bipartite graph structures by utilizing meta-paths. These extracted bipartite graphs are then projected and fused to transform them into a more simplified and synthesized graph structure. Finally, graph mining techniques are utilized to identify or filter the implicit learning partner relationships in the fused graph. The specific learning partner mining method is shown in Figure 3.

Figure 3 shows the process of projective fusion of bipartite maps as an example of two bipartite maps. If the learners u_i and u_j both learn the learning resource r_i, they are indirectly connected in the bipartite graph through the learning resource r_i. Therefore, the connected edges $(u_i, r_i), (u_j, r_i)$ in the bipartite graph are projected into the space where u is located, and u_i and u_j will be connected directly. And G_{B_1} and G_{B_2} correspond to two different bipartite graphs extracted from two different meta-paths, i.e., the learner nodes in the two graphs are the same while the learning resource nodes are different.

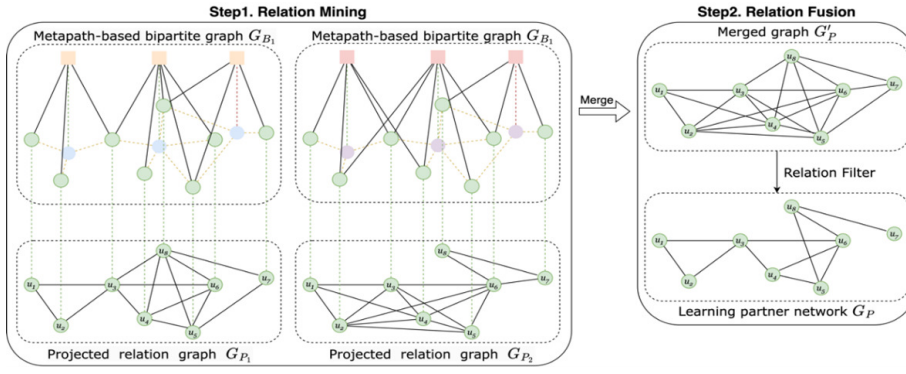


Figure 3. Learning partner mining process

Firstly, a single bipartite graph is projected and then the set of learner nodes connected to the same learning resource node constitutes a fully-connected subgraph. The fully-connected subgraphs corresponding to two different learning resources are connected by the learner who is learning these two learning resources at the same time. Finally, all the fully-connected subgraphs together constitute a homomorphic graph after projection. The projected homographs under metapaths need to be further fused, and Step2 in Figure 3 shows the specific fusion process. Since the weights of the edges of each node in the graph are different, and the higher the weight, the stronger the academic companionship, the edges of the nodes need to be further screened after weight fusion, i.e., only the edges with weights greater than the threshold value of σ produce the academic companionship relationship, which is formulated as:

$$A_{(}u_iu_j) = \begin{cases} A'_{u_iu_j}, & if A'_{u_iu_j} \geq \sigma, \\ 0, & \text{otherwise.} \end{cases} \tag{8}$$

The fused G_p contains only learner nodes and the connecting edges between the nodes represent the existence of a learning companion relationship between the learners, which is not only of some relevance, but also provides a basis for the subsequent community discovery task.

4.3.2 Community Discovery

The community discovery problem is a typical optimization problem and using intelligent optimization algorithms to improve the community discovery algorithm can improve the performance of the community discovery algorithm. So this paper proposes a community discovery algorithm based on the discrete beetle antennae algorithm for online learning companions. Beetle Antennae Search algorithm [32] is a single search intelligent optimization algorithm, which operates as follows: Assuming that the current position of the beetle is pos_t, then the positions of the left

and right beetle antennae are respectively $\text{pos}_t^l, \text{pos}_t^r$. The odor concentration perceived by the left and right beetle antennae needs to be compared at each iteration to determine the position pos_{t+1} where to move in the next iteration. The odor concentration used in the community discovery task is the size of the fitness, so the formula is expressed as:

$$\begin{cases} \text{pos}_t^l = \text{pos}_t - \frac{d_0}{2} \cdot \vec{d}, \\ \text{pos}_t^r = \text{pos}_t + \frac{d_0}{2} \cdot \vec{d}, \end{cases} \tag{9}$$

$$\text{pos}_{t+1} = \text{pos}_t + \sigma_t \cdot \vec{d} \cdot \text{sign}\left[\text{fit}\left(\text{pos}_t^r\right) - \text{fit}\left(\text{pos}_t^l\right)\right], \tag{10}$$

where t denotes the current number of iterations, and $\text{fit}(\cdot)$ denotes the fitness function, and σ_t denotes the length of the search step at the first iteration of t iteration, and $\text{sign}(\cdot)$ denotes the symbol function.

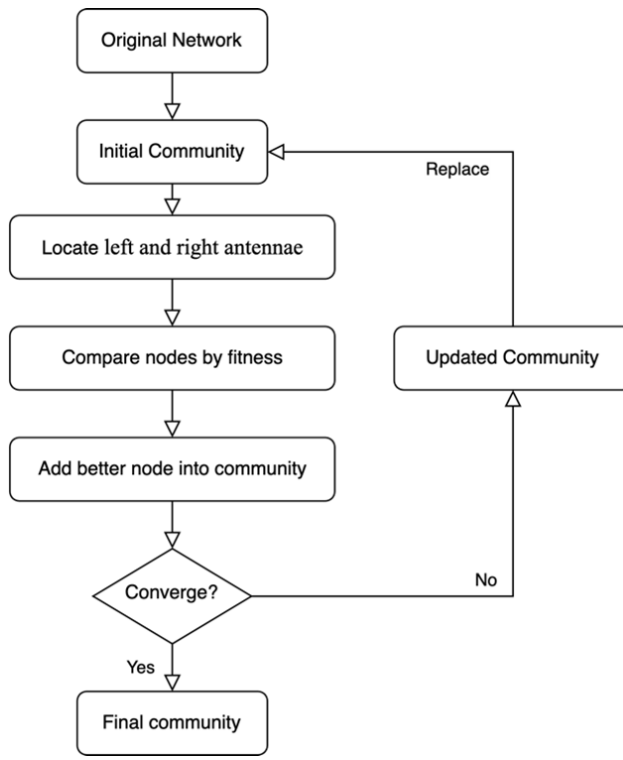


Figure 4. DBAS algorithm flow chart

However, the original beetle antennae search algorithm is a continuous process, to be used with the discrete community discovery, algorithm also needs to be discretized. The improved discrete beetle antennae algorithm (DBAS) flow is shown in Figure 4. The specific algorithm flow can be divided into the following three steps:

1. **Initial seed node screening.** The set of seed nodes in the learning companionship network is filtered using the k-shell decomposition algorithm. Iteratively remove the nodes in the network until all nodes are given k-shell values, and take the node with the largest k-shell value as the initial seed node, where seed_i denotes the first i seed node.
2. **Community Search.** Firstly, randomly locate the current beetle location pos_t in the neighborhood of the community where the seed node is located and then determine the positions $\text{pos}_t^l, \text{pos}_t^r$ of the left and right antennae by the current location pos_t. Initially, each seed node individually constitutes the community it belongs to. The formulaic expression of the location of the beetle and the left and right antennae is as follows:

$$\text{pos}_t = \text{Nei}(C_t^i)[rand(\cdot)], \tag{11}$$

$$\text{pos}_t^l = \text{Nei}(C_t^i)[0, \text{pos}_t], \tag{12}$$

$$\text{pos}_t^r = \text{Nei}(C_t^i)[\text{pos}_t, \text{len}(\text{Nei}(C_t^i)) - 1], \tag{13}$$

where $\text{Nei}(C_t^i)$ denotes the neighborhood of community C_t^i which the seed node seed_i belongs to and $rand(\cdot)$ is the random localization function. After determining the nodes where the left and right antennae positions of the beetle are located, the two nodes are respectively added to the community C_t^i to calculate fitness and then select the community structure with greater fitness to update the community. The fitness function used in this paper consists of community structure fitness and learning behavior similarity fitness, which is formulated as follows:

$$\text{fit}(C_t^i) = f_{stc}(C_t^i) + f_{sim}(C_t^i), \tag{14}$$

$$f_{stc}(C_t^i) = \sum_{C_t^i} d_{C_t^i}^{in} - \left(\frac{d_{C_t^i}^{in}}{d_{G_p}}\right)^2, \tag{15}$$

$$f_{sim}(C_t^i) = \sum_{u_i, u_j \in C_t^i} \text{pearson}(y_{u_i}, y_{u_j}), \tag{16}$$

where $f_{stc}(\cdot)$ denotes the community structure fitness function, and $f_{sim}(\cdot)$ denotes the learning behavior similarity fitness function; $d_{C_t^i}^{in}$ denotes the number of C_t^i's internal connected edges, and d_{G_p} denotes the number of C_t^i's external connected edges.
3. **Iterative computation.** The community structure will be updated in each iteration. In each iteration, C_t^i in the above process will be replaced by the updated community structure $C_t^{i\prime}$ and then proceed to another round of computation.

To further validate the algorithm's effectiveness, we conducted a time complexity analysis. Initially, k-shell decomposition identifies the initial community centers.

This phase's time complexity consists of two main components: the k-shell decomposition process and the node sorting process. The overall time complexity simplifies to $O(V \log V + E)$, where V denotes the number of nodes and E denotes the number of edges. Subsequently, the DBAS algorithm detects communities. In each iteration, the algorithm optimizes node placement across k communities. Each community evaluates fitness by incorporating the added node, incurring a primary time complexity of $O(n^2)$. Thus, the DBAS algorithm operates with a time complexity of $O(kn^2T)$, where T represents the number of iterations.

5 EXPERIMENT

In this section, we conduct experiments to demonstrate the effectiveness of the models in this paper. The dataset and baseline model used in this paper are introduced, and then the experimental results of the label prediction and community discovery tasks are shown and analyzed in detail.

5.1 Datasets

In this paper, we use IMDB[1], DBLP[2], and ZhihuRec datasets to train the label extraction model and use the model to predict the node labels in the ZhihuRec dataset and conduct community segmentation experiments based on it. The statistics of these three datasets and the metapath design of each dataset can be found in Table 2.

IMDB dataset. It is an online dataset on movies and TV shows, which includes three types of nodes: actors, production staff, and film and TV programs. The data-preprocessed dataset contains 4 278 movies, 2 081 directors, and 5 275 actors and actresses, and the nodes can be labeled as action, comedy, or drama based on the genre information.

DBLP dataset. It is a dataset collected by a computer science bibliography website, which includes four types of nodes: authors, papers, terms, and journals. The preprocessed dataset contains 4 057 authors, 14 328 papers, 7 723 terms, and 20 journals. Depending on the field of research, the author nodes can be labeled as data mining, database, artificial intelligence, and information retrieval fields.

ZhihuRec [33] dataset. It is a dataset collected from Zhihu, a public knowledge-sharing community, which allows the construction of high-quality questions and knowledge-sharing channels in the community through questions and answers between users, and users' behaviors such as querying and recommending exist in this scenario at the same time. After data preprocessing, the ZhihuRec dataset contains 7 963 users, 81 214 answers and 38 148 queries, and users are labeled into multiple categories based on the topics involved.

[1] `https://www.imdb.com/`

[2] `https://dblp.uni-trier.de/`

Dataset	Node	Edge	Metapath
ZhihuRec	user(U): 7 963 answer(A): 81 214 queries(Q): 38 148	U-A: 31 936 U-Q: 38 142	UAU, UQU, UAUAU, UQUQU, UAUQU, UQUAU
IMDb	movie(M): 4 278 director(D): 2 081 actor(A): 5 275	M-D: 4 278 M-A: 12 828	MDM, MAM, DMD, DMAMD, AMA, AMDMA
DBLP	author(A): 4 057 paper(P): 14 328 term(T): 7 723 venue(V): 20	A-P: 19 645 P-T: 85 810 P-V: 14 328	APA, APTPA, APVPA

Table 2. Dataset introduction and metapath design

In order to more intuitively describe the network structure of the online community and the characteristics of the learning behaviors of the learners it contains, the ZhihuRec dataset was selected in this paper to visualize the graph data construction process. As shown in Figure 5 a), different colors distinguish different types of nodes, and the colors from light to dark indicate learners, answer posts, and query requests. The more aggregated distribution of answer post resource nodes and learner resource nodes may be because the search keyword behavior is related to the learners' learning purpose, while the browsing behavior of answer posts is related to the platform's personalized push and service, and the answer posts with high view counts are usually recommended to learners. Figure 5 b) is a more intuitive description of the characteristics of the learning behavior of learners in the online K-S platform, according to the size of the node degree given different shades of color and size, the larger the node, the deeper the color indicates that the node's degree is higher. That is, the node represents a wider audience of learning resources, which is a darker node distribution compared with the distribution of nodes in Figure 5 a). It can be seen that the frequency of the same answer post to the learning behavior is significantly higher than the frequency of the search keywords, which is also the reason for the higher frequency of the answer post to the keyword. The reason for this is also related to the strength of learners' learning purpose for searching keywords and answering posts.

5.2 Baselines

Since the experiment is organized around two sub-problems:

1. how to obtain accurate interest labels for learning behaviors; and
2. how to divide the online learning partner community.

These two sub-problems correspond to different task objectives, for which different baseline models are set up, including GCN [26], GAT [28] and HAN [34], which are commonly used in the field of graph representation learning, and LFM [35],



a) Node distribution b) Node structure

Figure 5. Node network of online K-S platform

LPA [36] and GN [37] algorithms, which are commonly used in the field of community discovery.

Graph Representation Learning Baseline Model:

- GCN is a homogeneous graph neural network model that performs convolution operations in the Fourier domain of a graph, and in this paper, we use GCN on a network structure consisting of homogeneous metapaths.
- GAT is also a homogeneous graph neural network model that performs convolutional operations in the domain of the graph space containing the attention mechanism, and similarly to GCN, GAT is also used on network structures consisting of homogeneous metapaths.
- HAN is a heterogeneous graph neural network model that learns node embedding representations based on specific feature fusion methods from different metapath-based homogeneous compositions and combines them into a vector representation for each node by using an attention mechanism.

Community Discovery Baseline Model:

- LFM is a community partitioning algorithm based on seed node expansion, where the algorithm randomly selects nodes that have not been assigned a community ground as seeds and expands them to form a community structure by optimizing the fitness function.
- LPA is a local community partitioning algorithm based on label propagation, where each node in the network changes its label according to the labels of the nodes it is connected to, until the closely connected nodes in the network have a common label, which is fast and does not need to optimize any function

as compared to community discovery algorithms that optimize the degree of modularity.

- GN is a community discovery algorithm based on an optimization objective function that iteratively calculates the shortest path and the number of edge mediators for each edge in the network and removes the edge with the largest number of edge mediators to form a community structure, stopping the algorithm under certain conditions.

5.3 Experimental Results

In this section, we evaluate the model's performance from two key aspects. Initially, we confirm the effectiveness of the label prediction model. Subsequently, we employ this model to predict learner behavior, which forms the basis for the DBAS application. Finally, we utilize multiple experimental metrics to validate the efficacy of our community discovery algorithm.

5.3.1 Label Prediction

To evaluate the performance of label prediction models, we adopt several widely-used metrics in the classification task: F1 score, AUC and ACC. Table 3 shows the F1 score, AUC, and ACC values of this paper's label prediction model and the baseline model on the IMDB, DBLP, and ZhihuRec datasets, respectively, and it can be seen that our approach outperforms the GCN and GAT models by 1%–4%. The learning task of graph node representation using GCN and GAT does not perform semantic feature fusion, so the extracted node features are not very interpretable. Comparison experiments on the ZhihuRec dataset show that the heterogeneous neural network structure is superior for processing complex networks, especially heterogeneous information networks, and retaining the influence of the social network environment imposed by non-user nodes in the social network. The most important difference between online K-S platforms and traditional social networks is that the information generated from user interactions has educational attributes, i.e., it can have an impact on users' social behavior. When users have the identity of learners, the interaction information with educational attributes jointly constructs the online learning environment. Therefore, this paper employs heterogeneous graph neural networks to build the data structure of online K-S platforms and extract user features, demonstrating significant efficacy in predicting learner labels.

For several models on different heterogeneous social networks, performance differences are also greatly likely to be affected by the network structure. There is a significant difference in the model performance on IMDB compared with the other two datasets, which may be due to IMDB choosing a larger number of meta-paths, that is, the network structure is more complex. And this paper's preprocessing process for ZhihuRec dataset introduces some advantages. Even though ZhihuRec

Datasets	Metrics	GCN	GAT	HAN	Ours
IMDB	F1 score	52.80	53.64	56.32	**56.91**
	AUC	74.59	74.84	75.20	**75.88**
	ACC	80.64	80.98	81.57	**81.87**
DBLP	F1 score	88.51	91.61	92.33	**93.18**
	AUC	81.92	83.08	84.37	**85.13**
	ACC	77.82	78.36	79.41	**79.63**
ZhihuRec	F1 score	85.23	85.44	87.01	**87.58**
	AUC	68.26	68.45	69.34	**70.60**
	ACC	71.95	72.38	73.19	**73.29**

Table 3. Comparison results of label prediction

also designed many meta-path types, but the heterogeneous information network structure corresponding to each meta-path is not complex.

5.3.2 Community Detection

After verifying the effectiveness of the label prediction model, this paper uses the model to predict the behavior of user nodes in the ZhihuRec dataset and uses it for the final community discovery task. Table 4 demonstrates the performance of this paper's DBAS model with the baseline models LFM, LPA and GN model in terms of metrics such as fitness, modularity and behavioral similarity of nodes within the community. The DBAS algorithm proposed in this paper performs well in mining the structure of networked learning partner communities, and the similarity of learners' learning behavioral interests in the delineated cluster structure is also higher, DBAS has a better superiority in dealing with the detection of community structure on large-scale graph structures, and the mined community of networked learning partners also promotes the collaborative learning of similar learners to a certain extent.

Model	Metrics		
	Fitness	Modularity(Q)	Similarity
LFM	0.34	0.23	0.11
LPA	0.35	0.25	0.10
GN	0.40	0.28	0.12
DBAS	0.46	0.32	0.14

Table 4. Comparison results of K-S community detection

The DBAS model proposed in this paper performs well in the community discovery study of network learning partners, which may be due to the fact that the LFM algorithm and LPA algorithm are local community discovery algorithms with randomness, and thus are easy to fall into the local optimal situation, resulting in a low score of the fitness function. GN is a global community discovery algorithm,

but it is easily limited by the size of the network, and it is not applicable to the large-scale network structure. The DBAS proposed in this paper is also a locally extended community discovery algorithm. But it is able to jump out of the scope of local optimal search and thus easier to find the optimal community structure division.

In order to further validate the model robustness and efficiency of the DBAS method proposed in this paper on the community discovery task, comparative experiments are conducted on several commonly used datasets of social networks: Karate network, Dolphin network, OpenFlights Network [38], Football network [39] and Google Network [40]. By utilizing the modularity degree Q as well as the mutual information value NMI as evaluation metrics, the experimental results are presented in Tables 5 and 6. In Table 5, the performance of the DBAS method is compared with other commonly used community discovery methods on the Karate Network, Football Club Network and Dolphin Network datasets. The results show that the DBAS method outperforms the comparison models in most cases. However, DBAS is not optimal in terms of NMI metrics on the soccer club dataset and Q metrics on the dolphin dataset, which may be due to the fact that the DBAS method is more dependent on the initial topology of the network when selecting seed nodes. Especially when the network topology is more homogeneous, the community structure delineated by DBAS may appear sparse.

In addition, this paper validates the performance of the DBAS method on two large-scale social network datasets, namely, the Google Network and OpenFlights Network datasets. Through comparative experiments, the results demonstrate that the DBAS method is also able to achieve better results on these large-scale datasets, further proving its effectiveness and robustness on the community discovery task. These experimental results provide strong evidence for the DBAS approach in handling community discovery tasks on social network datasets of different sizes and types.

Datasets	Karate network		Football network		Dolphin network	
Metrics	Q	NMI	Q	NMI	Q	NMI
Louvain	0.3885	0.3951	0.6019	**0.7034**	**0.5100**	0.6081
CPM	0.1858	0.2215	0.3950	0.6382	0.3229	0.5795
SLPA	0.3494	0.3389	0.5971	0.6809	0.3140	0.5621
SCAN	0.2679	0.2876	0.5766	0.6935	0.3086	0.5059
CORPA	0.3594	0.3814	0.4903	0.6300	0.3794	0.5977
DBAS	**0.4668**	**0.4067**	**0.6100**	0.6736	0.4654	**0.6133**

Table 5. Comparison results of social community detection

The experiments show that the our model outperforms traditional GCN and GAT models by 1 %-4 % across various datasets due to its enhanced feature fusion capabilities. And it also examines the DBAS's performance in community detection within the ZhihuRec dataset, where it effectively identifies and enhances collaborative learning in networked learning communities. Additionally, the DBAS is tested

Datasets	Google network		OpenFlights network	
Metrics	Q	NMI	Q	NMI
Louvain	0.5414	0.6843	0.6362	0.4542
CPM	0.1288	0.4902	0.1426	0.4036
SCAN	0.2802	0.4759	0.3082	0.4233
CORPA	0.1428	0.5020	0.1959	0.4553
DBAS	0.5539	0.7025	0.6998	0.4613

Table 6. Comparison results of large-scale social community detection

on various network datasets including Karate, Dolphin, OpenFlights, Football, and Google networks, consistently outperforming other models, although it encounters some limitations on specific metrics in the Football and Dolphin datasets related to its dependency on initial network topology. Overall, the results underscore the DBAS model's robustness and effectiveness in community detection across different social network datasets.

6 CONCLUSION

In this paper, we first propose a model for label prediction of learners' learning behavioral interests and then decompose the learner relationship graph based on a bipartite graph to fill in the missing network structures. Finally, we propose a community discovery algorithm for network learning companions based on the discrete beetle antennae algorithm, which customizes the original beetle antennae search algorithm to adapt to the community discovery task. Experiments show that our model achieves better results compared to the baseline model in both label prediction and community discovery tasks, which to some extent verifies the feasibility of extracting complex heterogeneous information network structures based on bipartite graphs.

The optimization algorithms used in this paper are still limited to monolithic intelligent optimization algorithms. In the future, the use of population-based intelligent optimization algorithms can be considered to improve the work while using more dimensional optimization objectives to avoid falling into local optima.

Acknowledgements

This work was supported in part by the National Natural Science Foundation of China under Grant No. 62372300, Grant No. 62302306, and Grant No. 62201350, and in part by the National Key Research and Development Program of China under Grant No. 2022YFB4501704.

REFERENCES

[1] Kinshuk—Chen, N. S.—Cheng, I. L.—Chew, S. W.: Evolution Is Not Enough: Revolutionizing Current Learning Environments to Smart Learning Environments. International Journal of Artificial Intelligence in Education, Vol. 26, 2016, No. 2, pp. 561–581, doi: 10.1007/s40593-016-0108-x.

[2] Al-Kumaim, N. H.—Alhazmi, A. K.—Mohammed, F.—Gazem, N. A.—Shabbir, M. S.—Fazea, Y.: Exploring the Impact of the COVID-19 Pandemic on University Students' Learning Life: An Integrated Conceptual Motivational Model for Sustainable and Healthy Online Learning. Sustainability, Vol. 13, 2021, No. 5, Art. No. 2546, doi: 10.3390/su13052546.

[3] Narang, U.—Yadav, M. S.—Rindfleisch, A.: The "Idea Advantage": How Content Sharing Strategies Impact Engagement in Online Learning Platforms. Journal of Marketing Research, Vol. 59, 2022, No. 1, pp. 61–78, doi: 10.1177/00222437211017828.

[4] Chen, T.—Peng, L.—Jing, B.—Wu, C.—Yang, J.—Cong, G.: The Impact of the COVID-19 Pandemic on User Experience with Online Education Platforms in China. Sustainability, Vol. 12, 2020, No. 18, Art. No. 7329, doi: 10.3390/su12187329.

[5] Ivanova, A. D.—Murugova, O. V.: Online Education Through the Eyes of Students and Lecturers (Based on the Results of a Pedagogical Study in 2019). Open Education, Vol. 24, 2020, No. 2, pp. 4–16, doi: 10.21686/1818-4243-2020-2-4-16 (in Russian).

[6] Wen, J.—Wei, X.—He, T.—Zhang, S.: Regression Analysis on the Influencing Factors of the Acceptance of Online Education Platform Among College Students. Ingénierie des Systèmes d'Information, Vol. 25, 2020, No. 5, pp. 595–600, doi: 10.18280/isi.250506.

[7] Garrison, D. R.—Anderson, T.—Archer, W.: Critical Inquiry in a Text-Based Environment: Computer Conferencing in Higher Education. The Internet and Higher Education, Vol. 2, 1999, No. 2-3, pp. 87–105, doi: 10.1016/S1096-7516(00)00016-6.

[8] Lancichinetti, A.—Fortunato, S.: Limits of Modularity Maximization in Community Detection. Physical Review E, Vol. 84, 2011, Art. No. 066122, doi: 10.1103/PhysRevE.84.066122.

[9] Newman, M. E. J.: Modularity and Community Structure in Networks. Proceedings of the National Academy of Sciences, Vol. 103, 2006, No. 23, pp. 8577–8582, doi: 10.1073/pnas.0601602103.

[10] Girvan, M.—Newman, M. E. J.: Community Structure in Social and Biological Networks. Proceedings of the National Academy of Sciences, Vol. 99, 2002, No. 12, pp. 7821–7826, doi: 10.1073/pnas.122653799.

[11] Gao, G.—Cao, J.—Bu, Z.—Li, H. J.—Wu, Z.: A Generalized Game Theoretic Framework for Mining Communities in Complex Networks. Expert Systems with Applications, Vol. 96, 2018, pp. 450–461, doi: 10.1016/j.eswa.2017.10.058.

[12] Leskovec, J.—Lang, K. J.—Mahoney, M. W.: Empirical Comparison of Algorithms for Network Community Detection. Proceedings of the 19th International Conference on World Wide Web (WWW '10), ACM, 2010, pp. 631–640, doi: 10.1145/1772690.1772755.

[13] Li, H. J.—Bu, Z.—Li, A.—Liu, Z.—Shi, Y.: Fast and Accurate Mining the Community Structure: Integrating Center Locating and Membership Optimization. IEEE Transactions on Knowledge and Data Engineering, Vol. 28, 2016, No. 9, pp. 2349–2362, doi: 10.1109/TKDE.2016.2563425.

[14] Yuan, Q.—Liu, B.: Community Detection via an Efficient Nonconvex Optimization Approach Based on Modularity. Computational Statistics & Data Analysis, Vol. 157, 2021, Art. No. 107163, doi: 10.1016/j.csda.2020.107163.

[15] Hamilton, W.—Ying, Z.—Leskovec, J.: Inductive Representation Learning on Large Graphs. In: Guyon, I., Von Luxburg, U., Bengio, S., Wallach, H., Fergus, R., Vishwanathan, S., Garnett, R. (Eds.): Advances in Neural Information Processing Systems 30 (NIPS 2017). Curran Associates, Inc., 2017, pp. 1024–1034, doi: 10.48550/arXiv.1706.02216.

[16] Tang, X.—Yao, H.—Sun, Y.—Wang, Y.—Tang, J.—Aggarwal, C.—Mitra, P.—Wang, S.: Investigating and Mitigating Degree-Related Biases in Graph Convolutional Networks. Proceedings of the 29th ACM International Conference on Information & Knowledge Management (CIKM '20), 2020, pp. 1435–1444, doi: 10.1145/3340531.3411872.

[17] Zhang, T.—Xiong, Y.—Zhang, J.—Zhang, Y.—Jiao, Y.—Zhu, Y.: CommDGI: Community Detection Oriented Deep Graph Infomax. Proceedings of the 29th ACM International Conference on Information & Knowledge Management (CIKM '20), 2020, pp. 1843–1852, doi: 10.1145/3340531.3412042.

[18] Sun, X.—Zhang, W.—Guo, X.—Lu, W.: Unsupervised Graph Neural Network with Self-Expressive Attention for Community Detection. 2023 26th International Conference on Computer Supported Cooperative Work in Design (CSCWD), IEEE, 2023, pp. 1890–1895, doi: 10.1109/CSCWD57460.2023.10152715.

[19] Newman, M. E. J.—Girvan, M.: Finding and Evaluating Community Structure in Networks. Physical Review E, Vol. 69, 2004, No. 2, Art. No. 026113, doi: 10.1103/PhysRevE.69.026113.

[20] Blondel, V. D.—Guillaume, J. L.—Lambiotte, R.—Lefebvre, E.: Fast Unfolding of Communities in Large Networks. Journal of Statistical Mechanics: Theory and Experiment, Vol. 2008, 2008, No. 10, Art. No. P10008, doi: 10.1088/1742-5468/2008/10/P10008.

[21] von Luxburg, U.: A Tutorial on Spectral Clustering. Statistics and Computing, Vol. 17, 2007, No. 4, pp. 395–416, doi: 10.1007/s11222-007-9033-z.

[22] Murtagh, F.—Contreras, P.: Algorithms for Hierarchical Clustering: An Overview. WIREs Data Mining and Knowledge Discovery, Vol. 2, 2012, No. 1, pp. 86–97, doi: 10.1002/widm.53.

[23] Rosvall, M.—Bergstrom, C. T.: Maps of Random Walks on Complex Networks Reveal Community Structure. Proceedings of the National Academy of Sciences, Vol. 105, 2008, No. 4, pp. 1118–1123, doi: 10.1073/pnas.0706851105.

[24] Lancichinetti, A.—Fortunato, S.—Radicchi, F.: Benchmark Graphs for Testing Community Detection Algorithms. Physical Review E, Vol. 78, 2008, No. 4, Art. No. 046110, doi: 10.1103/PhysRevE.78.046110.

[25] Raghavan, U. N.—Albert, R.—Kumara, S.: Near Linear Time Algorithm to Detect Community Structures in Large-Scale Networks. Physical Review E, Vol. 76, 2007, No. 3, Art. No. 036106, doi: 10.1103/PhysRevE.76.036106.

[26] Kipf, T. N.—Welling, M.: Semi-Supervised Classification with Graph Convolutional Networks. 2017, doi: 10.48550/arXiv.1609.02907 (arXiv Preprint arXiv:1609.02907).

[27] Kipf, T. N.—Welling, M.: Variational Graph Auto-Encoders. 2016, doi: 10.48550/arXiv.1611.07308 (arXiv Preprint arXiv:1611.07308).

[28] Veličković, P.—Cucurull, G.—Casanova, A.—Romero, A.—Liò, P.—Bengio, Y.: Graph Attention Networks. 2018, doi: 10.48550/arXiv.1710.10903 (arXiv Preprint arXiv:1710.10903).

[29] Jin, D.—Liu, Z.—Li, W.—He, D.—Zhang, W.: Graph Convolutional Networks Meet Markov Random Fields: Semi-Supervised Community Detection in Attribute Networks. Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 33, 2019, No. 1, pp. 152–159, doi: 10.1609/aaai.v33i01.3301152.

[30] Wang, X.—Cui, P.—Wang, J.—Pei, J.—Zhu, W.—Yang, S.: Community Preserving Network Embedding. Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 31, 2017, No. 1, doi: 10.1609/aaai.v31i1.10488.

[31] Fu, X.—Zhang, J.—Meng, Z.—King, I.: MAGNN: Metapath Aggregated Graph Neural Network for Heterogeneous Graph Embedding. Proceedings of The Web Conference 2020 (WWW '20), ACM, 2020, pp. 2331–2341, doi: 10.1145/3366423.3380297.

[32] Jiang, X.—Li, S.: BAS: Beetle Antennae Search Algorithm for Optimization Problems. International Journal of Robotics and Control, Vol. 1, 2018, No. 1, pp. 1–5, doi: 10.5430/ijrc.v1n1p1.

[33] Hao, B.—Zhang, M.—Ma, W.—Shi, S.—Yu, X.—Shan, H.—Liu, Y.—Ma, S.: A Large-Scale Rich Context Query and Recommendation Dataset in Online Knowledge-Sharing. 2021, doi: 10.48550/arXiv.2106.06467 (arXiv Preprint arXiv:2106.06467).

[34] Wang, X.—Ji, H.—Shi, C.—Wang, B.—Ye, Y.—Cui, P.—Yu, P. S.: Heterogeneous Graph Attention Network. The World Wide Web Conference (WWW '19), 2021, pp. 2022–2032, doi: 10.1145/3308558.3313562.

[35] Chaudhary, L.—Singh, B.: Detecting Community Structures Using Modified Fast Louvain Method in Complex Networks. International Journal of Information Technology, Vol. 13, 2021, No. 5, pp. 1711–1719, doi: 10.1007/s41870-021-00790-z.

[36] Liu, Y.—Yu, J.—Liu, Z.—Han, X.: A Community Detection Algorithm Fusing Node Similarity and Label Propagation. In: Ma, H., Wang, X., Cheng, L., Cui, L., Liu, L., Zeng, A. (Eds.): Wireless Sensor Networks (CWSN 2022). Springer, Singapore, Communications in Computer and Information Science, Vol. 1715, 2022, pp. 94–107, doi: 10.1007/978-981-19-8350-4_8.

[37] Despalatović, L.—Vojković, T.—Vukičević, D.: Community Structure in

Networks: Girvan-Newman Algorithm Improvement. 2014 37th International Convention on Information and Communication Technology, Electronics and Microelectronics (MIPRO), IEEE, 2014, pp. 997–1002, doi: 10.1109/MIPRO.2014.6859714.

[38] Rossi, R.—Ahmed, N.: The Network Data Repository with Interactive Graph Analytics and Visualization. Proceedings of the AAAI Conference on Artificial Intelligence, Vol. 29, 2015, No. 1, doi: 10.1609/aaai.v29i1.9277.

[39] Kataoka, S.—Kobayashi, T.—Yasuda, M.—Tanaka, K.: Community Detection Algorithm Combining Stochastic Block Model and Attribute Data Clustering. Journal of the Physical Society of Japan, Vol. 85, 2016, No. 11, Art. No. 114802, doi: 10.7566/JPSJ.85.114802.

[40] Leskovec, J.—Lang, K. J.—Dasgupta, A.—Mahoney, M. W.: Community Structure in Large Networks: Natural Cluster Sizes and the Absence of Large Well-Defined Clusters. Internet Mathematics, Vol. 6, 2009, No. 1, pp. 29–123, doi: 10.1080/15427951.2009.10129177.



Bo Zhang received his Ph.D. degree in computer science from the College of Electronics and Information Engineering, Tongji University, Shanghai, China, in 2009. He finished his post-doctoral research work at the Tongji University in 2012. He is currently Professor with the College of Information, Mechanical and Electrical Engineering, Shanghai Normal University, Shanghai, and with Shanghai University of Electric Power, Shanghai. He is also the Director of the Social Network Analysis Project, which is funded by the National Natural Science Foundation of China. His current research interests include intelligent information processing, trust computation, and social network analysis.



Di Cai received his B.Sc. degree in computer science from the Xiamen University Malaysia, Malaysia, in 2022, and his M.Sc. degree in computer science from the Shanghai Normal University, Shanghai, China, in 2025. His current research interests include social network analysis and artificial intelligence in education.



Xinyi Zhang received her B.S. degree in computer science from Zhengzhou University, Zhengzhou, China, in 2019, and her M.Sc. degree in computer science from the Shanghai Normal University, Shanghai, China, in 2023. She is currently pursuing her Ph.D. degree at the Shanghai University, Shanghai, China. Her current research interests include social network analysis and artificial intelligence in education.



Hao Wang is currently a Vice President with the Shanghai Newtouch Software Co., Ltd., Shanghai, China. His work and research interests mainly focus on big data and artificial intelligence.



Zhonghua Zheng received his Ph.D. degree in management science and engineering from the University of Science and Technology of China, Hefei, China, in 2008. He is currently the Chairman of Anhui Boryou Information Technology Co., Ltd., Hefei, China, and he is a Senior Engineer. His work focuses on big data analysis, artificial intelligence, and data-driven information technology applications.



Meizi Li received her Ph.D. degree in computer science from the College of Electronics and Information Engineering, Tongji University, Shanghai, China, in 2021. She is currently Associate Professor with the College of Information, Mechanical and Electrical Engineering, Shanghai Normal University, Shanghai, China. Her current research interests include social network analysis, big data analysis and mining, and artificial intelligence education.



Ru Yang received her B.Sc. degree in computer science and technology from the Shandong University of Science and Technology, Qingdao, China, in 2013 and her Ph.D. degree in computer science and technology from the Tongji University, Shanghai, China, in 2020. She is currently Associate Professor with the College of Information, Mechanical and Electrical Engineering, Shanghai Normal University, Shanghai, China. Her current research interests include mobile systems, formal engineering and machine learning.