

PAPER • OPEN ACCESS

## Unsupervised community detection in attributed networks based on mutual information maximization

To cite this article: Junyou Zhu *et al* 2021 *New J. Phys.* **23** 113016

View the [article online](#) for updates and enhancements.

You may also like

- [Community detection in temporal networks via a spreading process](#)  
Peican Zhu, Xiangfeng Dai, Xuelong Li et al.
- [Leveraging disjoint communities for detecting overlapping community structure](#)  
Tanmoy Chakraborty
- [Self-falsifiable hierarchical detection of overlapping communities on social networks](#)  
Tianyi Li and Pan Zhang



## PAPER

# Unsupervised community detection in attributed networks based on mutual information maximization

## OPEN ACCESS

RECEIVED  
21 August 2021REVISED  
3 October 2021ACCEPTED FOR PUBLICATION  
14 October 2021PUBLISHED  
9 November 2021

Original content from  
this work may be used  
under the terms of the  
[Creative Commons  
Attribution 4.0 licence](#).

Any further distribution  
of this work must  
maintain attribution to  
the author(s) and the  
title of the work, journal  
citation and DOI.

Junyou Zhu<sup>1,2</sup>, Xianghua Li<sup>1,\*</sup>, Chao Gao<sup>1,2</sup>, Zhen Wang<sup>1,\*</sup> and Jurgen Kurths<sup>3,4</sup><sup>1</sup> School of Artificial Intelligence, Optics, and Electronics (iOPEN), Northwestern Polytechnical University, Xian 710072, People's Republic of China<sup>2</sup> College of Computer and Information Science & College of Software, Southwest University, Chongqing 400715, People's Republic of China<sup>3</sup> Potsdam Institute for Climate Impact Research, Potsdam, Germany<sup>4</sup> Faculty of Natural Sciences and Mathematics, University of Maribor, Koroška cesta 160, 2000 Maribor, Slovenia

\* Authors to whom any correspondence should be addressed.

E-mail: [li\\_xianghua@nwpu.edu.cn](mailto:li_xianghua@nwpu.edu.cn) and [zhenwang0@gmail.com](mailto:zhenwang0@gmail.com)**Keywords:** community detection, attributed networks, graph convolutional networks, mutual information

## Abstract

Community detection is of great significance for understanding network functions and behaviors. With the successful application of deep learning in network-based analyses, recent studies have turned to utilizing graph convolutional networks (GCNs) to this problem due to their capability in capturing network attributes. Nevertheless, most existing GCN-based community detection approaches are semi-supervised and local structure-aware, even though community detection is an unsupervised learning problem essentially. In this paper, we develop a novel GCN method for unsupervised community detection under the framework of mutual information (MI) maximization, called UCDMI. Specifically, a novel MI maximization mechanism is developed to capture more fine-grained information of the global network structure in an unsupervised manner. Moreover, a new aggregation function is proposed for GCN to distinguish the importance between different neighboring nodes, which enables our method to identify more high-quality node representations and improve the community detection performance. Our extensive experiments demonstrate the effectiveness of our proposed UCDMI compared with several state-of-the-art community detection methods.

## 1. Introduction

Community structures are common in complex networks, which can be described as some groups where nodes within the same group are closely connected [1–3]. The purpose of community detection is to detect such community structures from complex networks, which helps understand the hidden information of complex systems, e.g. the functions and units of a social group [4–6] and urban traffic systems [7, 8]. In reality, community detection owns many valuable applications and solves many practical problems, which have been widely focused in fields ranging from medicine and engineering to social science and biology [9–11]. For example, detecting community structure in World Wide Web networks can reveal different topics and facilitate social recommendations [12].

Up to now, numerous algorithms have been developed to detect communities by utilizing network structures, including generative models [13, 14] and metric-based methods [15, 16]. Besides the topology of networks, the effective utilization of attribute features also play a very important role in improving the accuracy of community detection [17]. Such attributes provide additional rich information of networks and indicate possible states of nodes in a network [18]. For example, in a citation network, papers are equipped with title and areas of keywords [19]. Since the topology structure and node attributes are two different types of information in networks, it poses a challenge to detect communities in such attributed networks.

Several methods have been proposed to detect communities of attributed networks by considering attributes of nodes, which include methods based on the nonnegative matrix factorization [6], spectral

clustering [20], graph convolutional networks (GCNs) [21], etc. Among them, GCN-based methods (e.g. DGI [22]) have recently gained a lot of attention due to their abilities to effectively integrate network topology information and attribute information. However, there are some issues which require further consideration. First, these methods typically update node features by aggregating the features of their neighboring nodes, which means treating all neighbors equally [23]. However, since different neighbor nodes have different positions in a network, their importances to the target node may also be different. Although the graph attention networks-based methods have considered this issue, they are implicit and may involve interpretability issue [24]. Second, most existing methods require some data with labels to train GCN. Nevertheless, the information of labels is often expensive or even unavailable due to the privacy policy [25]. Hence, it is crucial to develop a method that can detect high-quality community structures of networks in an unsupervised fashion. Third, most of these methods pay little attention to the global information of networks, which can be attributed to the fact that they are based on the local convolutional strategy [22]. Nevertheless, the global information may indicate the global status of nodes in a network, e.g. revealing the similarities between two nodes that are far away from each other but have similar connection patterns [26].

To fully inherit such global information and attributed features in an unsupervised manner, we execute community detection in attributed networks based on the mutual information (MI), inspired by the success of the Deep InfoMax method [27] on images processing. Deep InfoMax exploits global representations of an image by maximizing MI between images (i.e. the inputs) and hidden vectors (i.e. the outputs) [27]. Some recent works transfer Deep InfoMax to the network domain, such as deep graph infomax (DGI), for node classification and link prediction [22]. More specifically, they assume that the learned representations of each node should contain information of the entire graph. For this purpose, similar to the image processing, these methods discover useful node representations by implementing MI maximization based on the representation of the whole network. Nevertheless, learning the representation of nodes by capturing information from the entire network is crude and cannot reflect the intrinsic structural information. As mentioned before, there are community structures in networks, where nodes within the same community tend to share more similar information than nodes from different communities. In fact, for specific nodes, the information of their corresponding subnetworks (e.g. community structures) can better reflect their status and functions in the network than the information of the whole network. For example, a more accurate class information can better reflect the status of a specific student, such as the information of their major and grades, rather than a broader school information. However, existing methods which implement the MI maximization based on the whole graph cannot make use of this structure characteristic effectively, leading to rough node representations and efficiency decrease in community detection.

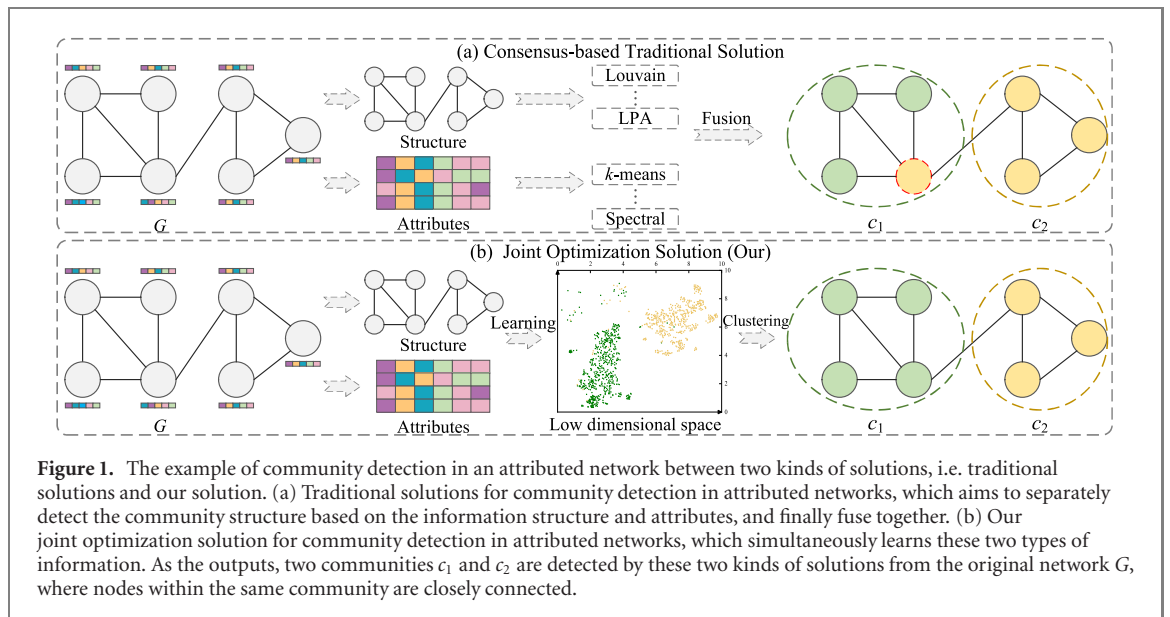
Considering these limitations, this paper proposes a new method for unsupervised community detection in attributed networks based on the MI maximization, called UCDMI. Specifically, we first analyze the potential cluster structure, namely fine-grained subnetworks, based on the attributed features of networks. Then, we develop a new MI maximization mechanism to maximize the MI between node representations and the corresponding fine-grained subnetwork representations. Based on such a mechanism, the more fine-grained information of the global network structure and attributes can be effectively captured in an unsupervised manner. Moreover, we design a new aggregation function of GCN that can aggregate the features of neighbors in preference according to the importance between nodes. Finally, the community structure of networks can be obtained by applying a clustering algorithm on node embedding. The main contributions of this paper are as follows:

- (a) A novel aggregation function for GCN is proposed in which more important neighbor nodes can contribute more in the process of the feature aggregation.
- (b) A new MI maximization mechanism is designed in which more fine-grained global information of networks is captured in an unsupervised fashion, which helps get high-quality community structures in attributed networks.

The remainder of this paper is organized as follows. Section 2 reviews the related work. Section 3 introduces the proposed UCDMI, which describes the formulations of the proposed aggregation strategy and the new MI maximization mechanism. Section 4 presents comprehensive experiments to demonstrate the effectiveness of UCDMI. Finally, section 5 draws the conclusions of this paper.

## 2. Related work

Existing community detection methods related to this work fall into two categories: structural community detection and attributed community detection [28]. Structural community detection only utilizes the structure of networks (i.e. node connectivity [29, 30]). Specifically, the graph Laplacian eigenmaps-based



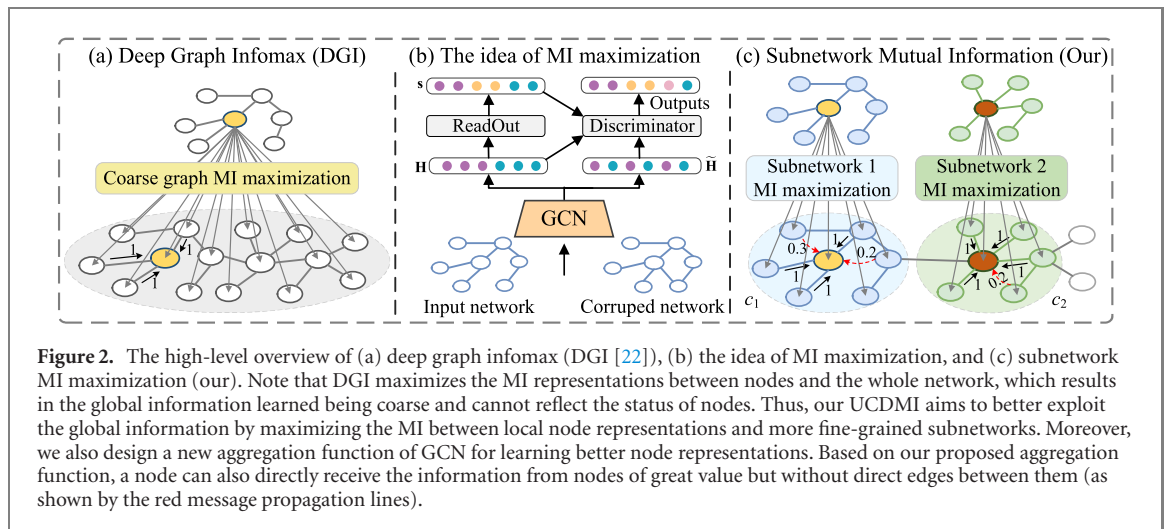
**Figure 1.** The example of community detection in an attributed network between two kinds of solutions, i.e. traditional solutions and our solution. (a) Traditional solutions for community detection in attributed networks, which aims to separately detect the community structure based on the information structure and attributes, and finally fuse together. (b) Our joint optimization solution for community detection in attributed networks, which simultaneously learns these two types of information. As the outputs, two communities  $c_1$  and  $c_2$  are detected by these two kinds of solutions from the original network  $G$ , where nodes within the same community are closely connected.

methods assume that similar nodes should be mapped closer [31]. To exploit more structural relationships, the matrix factorization-based methods factorize the adjacency matrix and other relationship matrices of networks into node representations [32]. For example, a recent method proposes an NMF-based method to detect community structures by solving the majorization-minimization principle [2]. However, the matrix factorization has high time complexity due to the frequent matrix operations. Therefore, the random walk-based methods are used instead to maximize the probability of neighborhoods for each node to overcome this problem [33]. Moreover, although the autoencoder-based methods can also exploit relationships of the network structure by using the learned node representations to reconstruct the adjacency matrix, these methods only exploit network structures [34]. Besides, some methods, e.g. NSGAMOF, apply the modularity-based technology for community detection in order to further exploit the structures [35]. However, some attributes that can provide extra profiles for users should be taken into account.

Different from structural community detection methods, attributed community detection methods consider both the structures and attributes of networks [36]. Specifically, some existing methods aim to fuse the information of structure and attributes after the process of community detection. As shown in figure 1(a), these methods first separately detect communities based on the attributes (e.g. by  $k$ -means [37, 38]) and the structure (e.g. by Louvain [39]). Then, the resulting communities are merged in a way that results in structure-and attributes-aware communities. For example, CFOND utilizes the consensus factorization principle to preserve the information of the structure and attributes for co-clustering network data. Similarly, based on the consensus clustering [40], FCCCN can be applied to networks with millions of nodes by calculating the consensus matrix and additional node pairs [41]. In fact, the information of the structure and attribute are interrelated, and considering the two types of information separately during the process of community detection may sever the connection between the both [6].

In order to jointly optimize both kinds of information, as shown in figure 1(b), some recent methods have applied deep learning-based methods on the attributed and large-scale networks to detect underlying community structure [42]. Specifically, vGraph detects communities in attributed networks by utilizing a generative model to jointly learn node attributes and network structures [43]. However, although vGraph has the ability to capture node attributes, it mainly relies on the structural information of networks when computing the scores of community distribution over nodes [25]. To make full use of the attribute information, some methods (e.g. ARVGE [44] and DAEGC [25]) use a graph autoencoder to encode the node attributes and topological structure of a network into a compact representation. To some extent, these methods revealing community structures depend on adjacency matrix reconstruction, which splits the intrinsic relationship between node attributes and network structure [45]. To overcome this problem, some recent GCN-based methods (e.g. GUCD [42]) use the structural information (often denoted as adjacency matrix) to guide the aggregation of node attributes, thus effectively unifying these two kinds of information to better detect communities in networks.

However, existing GCN-based methods do not address the global information of networks because they adopt the local aggregation strategy [21]. To overcome this, DGI proposes a MI maximization strategy to embed the global information into node representations [22]. Nevertheless, the global information captured



**Figure 2.** The high-level overview of (a) deep graph infomax (DGI [22]), (b) the idea of MI maximization, and (c) subnetwork MI maximization (our). Note that DGI maximizes the MI representations between nodes and the whole network, which results in the global information learned being coarse and cannot reflect the status of nodes. Thus, our UCMI aims to better exploit the global information by maximizing the MI between local node representations and more fine-grained subnetworks. Moreover, we also design a new aggregation function of GCN for learning better node representations. Based on our proposed aggregation function, a node can also directly receive the information from nodes of great value but without direct edges between them (as shown by the red message propagation lines).

by DGI is coarse, since it maximizes MI based on the whole network. Therefore, how to develop a novel method that can capture both the fine-grained global information and node attributes is still an open problem.

### 3. The formulation of UCMI

In this section, we first give the notations and problem definition, and then present our proposed UCMI. Specifically, as shown in figure 2(c), there are two key parts of UCMI: the new aggregation strategy of GCN and the novel MI maximization mechanism. The new aggregation strategy can adjust the contributions of nodes in the process of feature aggregation according to the importance between nodes. As shown in the red message propagation lines of figure 2(c), the information of features can even pass directly between valuable nodes where no direct edges exist between them. Moreover, different from existing MI maximization-based methods (e.g. DGI, which is shown in figure 2(a)), our proposed MI maximization mechanism can much better capture the global information of networks by maximizing the MI between local node representations and more fine-grained subnetworks. More differences between our approach and DGI will be described in detail later.

#### 3.1. Notations and problem definition

We formally define an undirected and attributed network as  $G = (V, E, \mathbf{X})$ , where  $V = \{v_1, \dots, v_n\}$  is a set of nodes, and  $E = \{e_{ij}\}_{i,j=1}^V$  represents a set of edges.  $\mathbf{X} \in \mathbb{R}^{n \times f}$  is a feature matrix for all nodes, where  $f$  means the number of features of a node. Given an attributed network  $G$ , the aim of community detection is to find some groups where nodes within the same group are closely connected. Formally, such groups are referred as communities, i.e.  $C = \{c_1, \dots, c_m\}$ .

#### 3.2. Graph convolutional layer

To realize the MI maximization, we should capture the node-level information, i.e. node representations. In this work, our encoder  $\mathcal{E}_{\text{our}}$  is based on the GCNs. The original propagation rule  $\mathcal{E}_{\text{GCN}}$  of GCN is defined as

$$\mathbf{H} = \mathcal{E}_{\text{GCN}}(\mathbf{X}, \mathbf{A} | \mathbf{W}) = \sigma \left( \hat{\mathbf{D}}^{-\frac{1}{2}} \hat{\mathbf{A}} \hat{\mathbf{D}}^{-\frac{1}{2}} \mathbf{X} \mathbf{W} \right), \quad (1)$$

where  $\mathbf{A}$  is an adjacency matrix of a network  $G$ ,  $\hat{\mathbf{A}} = \mathbf{A} + \mathbf{I}_n$  represents the summation of the identity matrix and adjacency matrix.  $\hat{\mathbf{D}}$  is a degree matrix where  $\hat{D}_{i,i} = \sum_j \hat{A}_{i,j}$ . The  $\sigma$  is the ReLU function, and  $\mathbf{W}$  is a learnable linear transformation.  $\mathbf{H} = \{\mathbf{h}_1, \dots, \mathbf{h}_n\}$  denotes the matrix of node representations in which  $h_i \in \mathbb{R}^d$  is a low-dimensional vector of  $v_i$ , and  $d$  means the number of embedding dimensions.

Based on equation (1), the propagation rule of GCN is to aggregate features of neighboring nodes in average, since it uses  $\mathbf{A} + \mathbf{I}_n$  to guide the aggregation, i.e. all neighboring nodes are treated equally. Besides, some valuable nodes that are important to each other but not directly connected cannot share the information directly. However, in the real world, the importance of different friends to a person often varies according to the positions of these friends in society, and some indirect friends may also be important. This motivates us to design an effective propagation rule for guiding the aggregation according to the importance between nodes.

To this end, we quantify the importance between nodes by measuring the Jaccard similarity of nodes because of its empirical success in graph clustering and node classification. Specifically, given two nodes  $v_i$  and  $v_j$ , the Jaccard similarity between them, i.e.  $S_{v_i, v_j} \in \mathbf{S}$ , is calculated as

$$S_{v_i, v_j} = \frac{|N(v_i) \cap N(v_j)|}{|N(v_i) \cup N(v_j)|}, \quad (2)$$

where  $N(v_i)$  and  $N(v_j)$  represent a set of neighbors of  $v_i$  and  $v_j$ , respectively. Such similarity between two nodes indicates how important they are to each other, and will be used to guide the aggregation. Therefore, based on the Jaccard similarity matrix  $\mathbf{S}$ , we design our encoder  $\mathcal{E}_{\text{our}}$  as

$$\mathbf{H} = \mathcal{E}_{\text{our}}(\mathbf{X}, \mathbf{S}, \mathbf{A}|\mathbf{W}) = \sigma\left(\hat{\mathbf{D}}^{-\frac{1}{2}} \hat{\mathbf{S}} \hat{\mathbf{D}}^{-\frac{1}{2}} \mathbf{X} \mathbf{W}\right), \quad (3)$$

where  $\hat{\mathbf{S}} = \mathbf{A} + \mathbf{I}_n + \alpha \mathbf{S}$ , and  $\alpha$  is a parameter used to balance the contribution of the Jaccard similarity. According to the encoder we designed, a node no longer aggregates the features of neighboring nodes equally, but adopts an unequal aggregation based on the values of  $\hat{\mathbf{S}}$  instead. The information about nodes that are important to each other but not directly connected is also taken into account.

### 3.3. Mutual information maximization mechanism

Since GCN-based frameworks utilize the local propagation rule, this results in the node representations learned by equation (3) retaining only the local information of networks. Thus, we proposed a new MI maximization mechanism that enable nodes embedding to capture the global information of networks. It is realized in three steps: (1) finding some fine-grained subnetworks that indicate potential clusters; (2) computing the global representations of such subnetworks; (3) maximizing the MI between the global presentations of each subnetwork and the local representations of nodes residing in the subnetwork.

As mentioned before, different from existing MI maximization-based methods (e.g. DGI) which design MI based on the whole graph representations, we use some more fine-grained subnetworks instead. Given a graph  $G$  with the attributed matrix  $\mathbf{X}$ , we aim to design a function  $\mathcal{P}(\mathbf{X})$  to find some subnetworks indicating potential clustering. In this paper, we adopt  $k$ -means to serve as such function because it works best in experiments, which can be defined as

$$\{g_1, \dots, g_m\} = \mathcal{P}(\mathbf{X}) = k \text{ means}(\mathbf{X}), \quad (4)$$

where  $g_i$  is a fine-grained subnetwork. Having trained the local node representations in equation (3) and a set of subnetworks  $\{g_1, \dots, g_m\}$ , we introduce a subnetwork-level representation encoder, i.e. a readout function:  $\mathcal{R}: \mathbb{R}^{p \times d} \rightarrow \mathbb{R}^d$ . To avoid notational cluttering, we assume that there are  $p$  nodes in each subnetwork, although we reiterate that the number of nodes in each subnetwork may be different. Specifically, for each subnetwork, we aim to obtain its global representations  $\mathbf{s}_r^t$ , where  $1 \leq r \leq m$ . Formally, we define such a readout function as

$$\mathbf{s}_r^t = \mathcal{R}(\mathbf{H}_r) = \sigma\left(\frac{1}{p} \sum_{i=1}^p \mathbf{h}_i^r\right), \quad (5)$$

where  $\mathbf{H}_r \in \mathbb{R}^{p \times d}$  represents the corresponding node representation matrix of subnetwork  $g_r$  extracted from  $\mathbf{H}$ , and  $\mathbf{h}_i^r$  is the  $i$ th row in  $\mathbf{H}_r$ .  $\sigma$  denotes the logistic sigmoid nonlinearity.

Finally, to enable our model to capture more fine-grained global information, for each subnetwork  $g_r$ ,  $1 \leq r \leq m$ , we maximize the MI between the local node representation patches  $\{\mathbf{h}_1^r, \mathbf{h}_2^r, \dots, \mathbf{h}_p^r\}$  and the corresponding summary network representations  $\mathbf{s}_r$ . For this purpose, we use a discriminator  $\mathcal{D}$  which discriminates true samples [i.e.  $(\mathbf{h}_i^r, \mathbf{s}_r^t)$ ] from its negative samples [i.e.  $(\tilde{\mathbf{h}}_j^r, \mathbf{s}_r^t)$ ] and the final loss is defined as

$$\mathcal{L}_{\text{loss}} = \sum_{i=1}^n \log \mathcal{D}(\mathbf{h}_i^r, \mathbf{s}_r^t) + \sum_{j=1}^n \log \left(1 - \mathcal{D}(\tilde{\mathbf{h}}_j^r, \mathbf{s}_r^t)\right), \quad (6)$$

where the negative node representations  $\tilde{\mathbf{h}}_j^r$  correspond to the  $j$ th row in  $\tilde{\mathbf{H}}_r$ .  $\mathcal{D}$  is a discriminator used to score the pairs  $(\mathbf{h}_i^r, \mathbf{s}_r^t)$ . In this paper, the simple bilinear scoring function is utilized, as equation (7).

$$\mathcal{D}(\mathbf{h}_i^r, \mathbf{s}_r^t) = \sigma(\mathbf{h}_i^{rT} \mathbf{B}^t \mathbf{s}_r^t), \quad (7)$$

where  $\mathbf{B}^t$  denotes a trainable scoring matrix, and  $\sigma$  represents the logistic sigmoid nonlinearity.  $\tilde{\mathbf{H}}_r$  is extracted from  $\tilde{\mathbf{H}}$  in the same way as  $\mathbf{H}_r$ . To construct the negative node representation matrix  $\tilde{\mathbf{H}}$ , we first shuffle the original attribute matrix  $\mathbf{X}$  in the row-wise fashion to generate the corrupted network, i.e.

**Algorithm 1.** The UCDMI algorithms.

---

**Input:** an undirected and attributed network  $G = (V, E, \mathbf{X})$   
**Output:** a set of communities  $C = \{c_1, \dots, c_m\}$

- 1 Initialize trainable parameters  $\mathbf{W}$
- 2 Constructing  $\mathbf{S}$ ; // calculate the Jaccard similarity based on equation (2)
- 3 Mining the fine-grained subnetworks  $\{g_1, \dots, g_m\}$  based on equation (4)
- 4 **while** *convergence* **do**
- 5      $\mathbf{H} = \mathcal{E}_{\text{our}}(\mathbf{X}, \mathbf{S}, \mathbf{A} \mid \mathbf{W})$  // learning local node representations (positive)
- 6      $\mathbf{X} \rightarrow \tilde{\mathbf{X}}$  // obtain  $\tilde{\mathbf{X}}$  by shuffling  $\mathbf{X}$
- 7      $\tilde{\mathbf{H}} = \mathcal{E}_{\text{our}}(\tilde{\mathbf{X}}, \mathbf{S}, \mathbf{A} \mid \mathbf{W})$  // learning local node representations (negative)
- 8     **foreach**  $r \in m$  **do**
- 9         Getting the  $\mathbf{H}_r$  for each subnetwork  $g_r$  from  $\mathbf{H}$
- 10          $\mathbf{s}_r^t = \mathcal{R}(\mathbf{H}_r)$  // calculate the global representations of  $g_r$
- 11         Compute the cross entropy, i.e.,  $\mathcal{L}_{\text{loss}}$  (equation (6))
- 12     Obtain communities  $C = \{c_1, \dots, c_m\}$  by applying  $k$ -means on  $\mathbf{H}$

---

$\mathbf{X} \rightarrow \tilde{\mathbf{X}}$ . Then, we reuse the encoder defined in equation (3) to generate  $\tilde{\mathbf{H}}$ , that is,  $\tilde{\mathbf{H}} = \mathcal{E}_{\text{our}}(\tilde{\mathbf{X}}, \mathbf{S}, \mathbf{A} \mid \mathbf{W})$ . It is remarkable that, our novel MI maximization mechanism amounts to the binary cross entropy loss shown in equation (6). Such theory has been proved in previous works [22], and also fits with our mechanism, although the details of our model differ from theirs. By minimizing the loss of equation (6) based on the designed GCN, i.e. equation (3), we obtain the node representations which capture the fine-grained global information of networks. Based on such node representations, the community structures are identified by directly applying a clustering algorithm on it, such as  $k$ -means. Algorithm 1 shows the pseudocode of UCDMI.

### 3.4. Complexity analysis

The time complexity of UCDMI depends on the operation of convolution in GCN and the calculation of the Jaccard similarity. Concretely, according to [46], the time complexity involved in GCNs is  $O(|E|)$ . It means that the time complexity of the convolution operation increases linearly with the number of edges.

In terms of calculating the Jaccard similarity, the complexity of calculating two sets  $X$  and  $Y$  with  $n$  elements is  $O(n \log n)$ . For calculating the similarity of nodes in complex networks, we measure the similarity between target nodes and their corresponding neighboring nodes. Let  $d_{\text{avg}}$  denote the average degree of a node. Then, the calculation of the Jaccard similarity between a pair of nodes has the complexity  $O(|E|d_{\text{avg}} \log d_{\text{avg}})$ , where  $|E|$  represents the number of edges. Thus, the overall time complexity of UCDMI is  $O(|E|d_{\text{avg}} \log d_{\text{avg}})$ .

## 4. Experiments

In this section, we first introduce the datasets, baseline methods, and parameter setting involved in this paper. Then, we compare the performance of our proposed UCDMI with other baseline methods on community detection. Moreover, we also provide some further investigation, i.e. ablation study, parameter analysis, and visualization to understand the effectiveness of the proposed strategies.

### 4.1. Datasets and evaluation metrics

#### 4.1.1. Datasets

Several widely used and standard networks are analyzed to verify the effectiveness of our proposed methods. A summary of these networks is presented in table 1, and the detailed information of these methods is introduced as follows.

**Polbooks network**<sup>5</sup>: Polbooks network reflects the sales relationship of political books on Amazon, where the nodes represent the books sold on Amazon, and each edge between two nodes means that the two books are purchased by the same customer. Since the attributes of nodes in this network are not provided, we use the vectors extracted from the adjacency as such attributes.

<sup>5</sup> <http://www-personal.umich.edu/mejn/netdata/>.

**Table 1.** The tested attributed networks for the experiments.

	Dataset	Nodes	Edges	Communities	Features
$G_1$	Polbooks	105	441	3	105
$G_2$	Cora	2708	5429	7	1433
$G_3$	Citeseer	3327	4732	6	3703
$G_4$	Pubmed	19 717	44 378	3	500
$G_5$	Parliament	451	11 646	2	108
$G_6$	Wiki	2405	17 981	17	4973
$G_7$	Synthetic	30 000	117 600	8	1

**Cora network<sup>6</sup>:** Cora network is built based on the citation relationship between machine learning papers, where the nodes represent published papers, and edges denote the citation relationship between two papers.

**Citeseer network<sup>6</sup>:** Citeseer network is also built based on the citation relationship between scientific papers, where the nodes represent published papers, and edges denote the citation relationship between two papers.

**Pubmed network<sup>6</sup>:** Pubmed network describes the citation relationships between diabetes-related scientific papers in Pubmed database, where the nodes represent published papers, and edges denote the citation relationship between two papers.

**Parliament network<sup>7</sup>:** Parliament network is a bill co-signer network where the nodes denote French parliament members. If two parliament members jointly sign a bill, there is an edge between them.

**Wiki network<sup>8</sup>:** Wiki network describes the links between webpages, where the nodes represent webpages and edges mean the links between webpages.

**Synthetic network<sup>9</sup>:** Synthetic network is the benchmark generated by Elhadi and Agam [47], and all the settings of parameters come from [48].

#### 4.1.2. Evaluation metrics

To evaluate the performance of UCDMI and baselines in community detection, we adopt three typical and widely used metrics: normalized mutual information (*NMI*), clustering accuracy (*ACC*), and macro *F1*-score (*F1*). The higher the value of these metrics, the better the performance of community detection.

#### 4.2. Parameter setting and baselines

In UCDMI, taking the efficiency into account, the node representation dimension  $d$  is set to 128, and the number of layer in the GCN is set to 1. For the parameter  $\alpha$ , which balances the contribution of the proposed aggregation strategy, we set  $\alpha = \{0.2, 0.3, 0.5\}$ . The parameter analysis is given in section 4.5.

To fairly evaluate the performance of UCDMI when dealing with community detection on attributed networks, we compare our proposed method with three kinds of community detection methods, and the parameters of these methods are consistent with the source papers. Concretely, there are methods that only use node attributes, methods that only use network structure, and methods that use the both. We list these baseline methods as follows.

- Methods that only use node attributes:**  $k$ -means [37] and spectral clustering [20] (Spectral-a) methods which detect community structures only based on node attributes;
- Methods that only use network structure:** spectral clustering (spectral-s) that uses the adjacency of networks as its inputs, DeepWalk [33], and DNGR [34];
- Methods that use both the network structure and attributes:** community embedding framework (ComE) [17], adversarially regularized graph autoencoder (ARVGE) [44], probabilistic generative model (vGraph) [43], deep attention embedded graph clustering (DAEGC) [25], DGI [22], and unsupervised GCN-based method for community detection (GUCD) [42].

#### 4.3. Results of community detection

The *NMI*, *ACC*, and *F1*-score of UCDMI and other baseline methods on benchmark networks are reported in tables 2 and 3. There are several key observations. Compared with baseline methods, UCDMI achieves relatively good results on all experimental datasets. Specifically, compared to methods that use either node

<sup>6</sup> <https://github.com/tkipf/gcn/tree/master/gcn/data>.

<sup>7</sup> <https://github.com/abojchevski/paican/tree/master/data/parliament>.

<sup>8</sup> <https://github.com/karenlatong/AGC-master>.

<sup>9</sup> <http://networkrepository.com/SYNTHETIC.php>.



**Table 2.** Results of community detection by UCDDMI and other baseline methods in  $G_2$ ,  $G_3$ , and  $G_4$ . The best results are recorded in bold. Results show that our proposed method is better than other baselines.

Methods	Input	$G_2$			$G_3$			$G_4$		
		NMI	ACC	F1	NMI	ACC	F1	NMI	ACC	F1
$k$ -means	Attributes	0.167	0.347	0.254	0.170	0.385	0.305	0.291	0.573	0.574
Spectral-a	Attributes	0.151	0.363	0.256	0.212	0.462	0.337	<b>0.325</b>	0.599	0.586
Spectral-s	Structure	0.195	0.342	0.302	0.118	0.259	0.295	0.147	0.496	0.471
DeepWalk	Structure	0.316	0.467	0.381	0.131	0.390	0.305	0.167	0.619	0.471
DNGR	Structure	0.373	0.492	0.373	0.180	0.326	0.442	0.154	0.454	0.179
ComE	Both	0.366	0.438	0.435	0.250	0.441	0.448	0.300	0.415	0.552
ARVGE	Both	0.450	0.638	0.627	0.261	0.544	0.529	0.206	0.582	0.230
vGraph	Both	0.345	0.287	0.305	0.103	0.293	0.294	0.224	0.260	0.332
DAEGC	Both	0.528	0.704	0.682	0.397	0.672	0.636	0.266	0.671	0.659
DGI	Both	0.457	0.651	0.566	0.431	0.672	<b>0.655</b>	0.288	0.608	0.599
GUCD	Both	0.323	0.505	0.568	0.274	0.545	0.527	0.269	0.631	0.604
<b>UCDDMI</b>	Both	<b>0.573</b>	<b>0.732</b>	<b>0.715</b>	<b>0.441</b>	<b>0.694</b>	0.649	<b>0.307</b>	<b>0.692</b>	<b>0.688</b>

**Table 3.** Results of community detection by UCDDMI and other baseline methods in  $G_5$ ,  $G_6$ , and  $G_7$ . The best results are recorded in bold. Results show that our proposed method is better than other baselines.

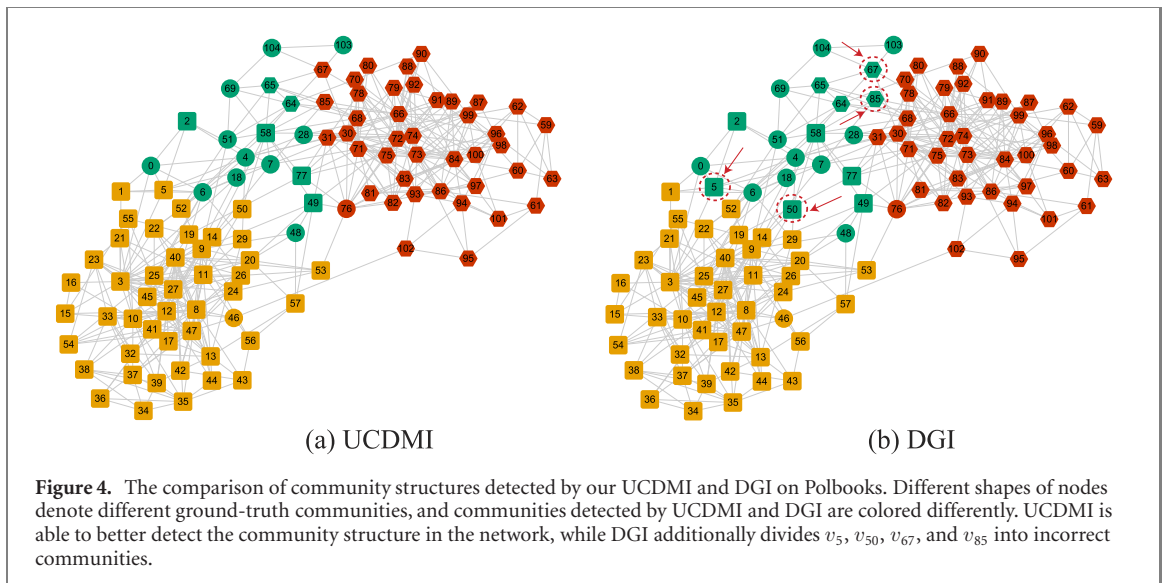
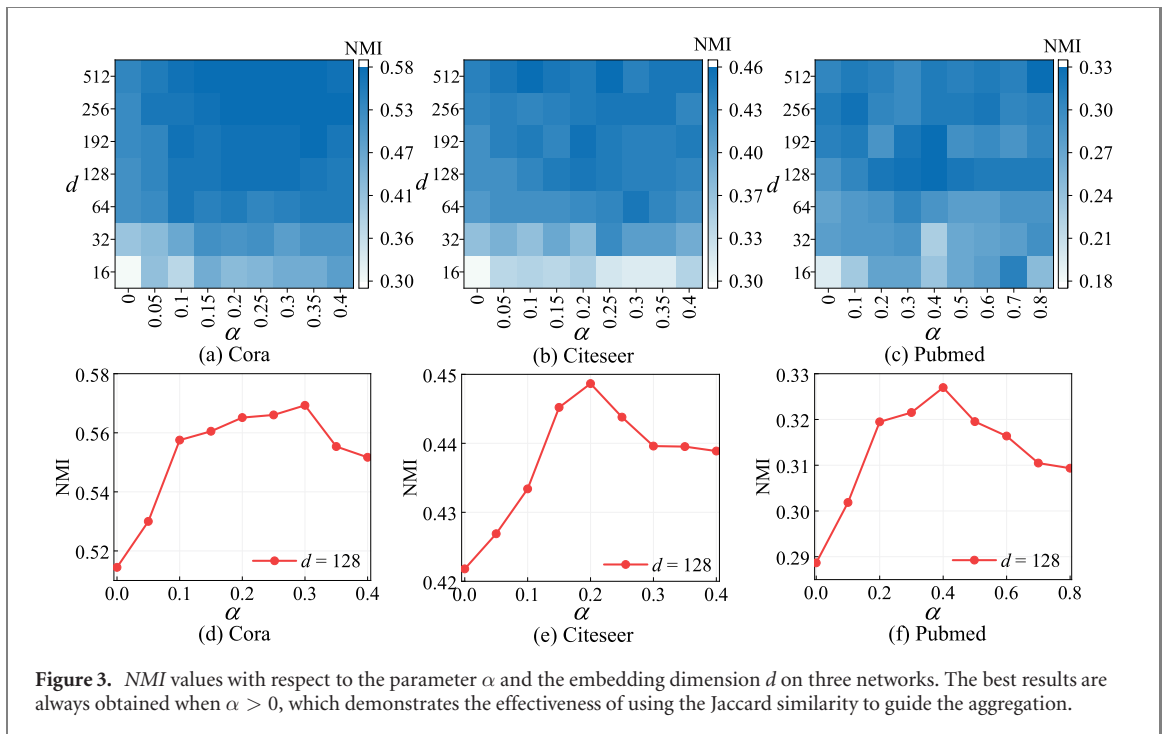
Methods	Input	$G_5$			$G_6$			$G_7$		
		NMI	ACC	F1	NMI	ACC	F1	NMI	ACC	F1
$k$ -means	Attributes	0.042	0.365	0.151	0.206	0.261	0.121	0.043	0.199	0.172
Spectral-a	Attributes	0.043	0.376	0.171	0.015	0.168	0.020	0.178	0.261	0.148
Spectral-s	Structure	0.067	0.361	0.088	0.185	0.242	0.145	0.242	0.329	0.176
DeepWalk	Structure	0.603	0.741	0.254	0.351	0.429	0.318	0.012	0.142	0.117
DNGR	Structure	0.327	0.289	0.425	0.276	0.299	0.197	0.382	0.276	0.332
ComE	Both	0.477	0.516	0.584	0.359	0.337	0.273	0.482	0.414	0.453
ARVGE	Both	0.657	0.746	0.519	0.308	0.274	0.357	0.317	0.468	0.399
vGraph	Both	0.482	0.618	0.424	0.274	0.326	0.276	0.410	0.465	0.387
DAEGC	Both	0.548	0.741	0.255	0.241	0.296	0.254	0.482	0.327	0.319
DGI	Both	0.562	0.633	0.478	0.457	0.461	<b>0.401</b>	0.429	0.443	0.405
GUCD	Both	0.618	0.729	0.624	0.321	0.385	0.304	0.419	0.438	0.271
<b>UCDDMI</b>	Both	<b>0.729</b>	<b>0.821</b>	<b>0.731</b>	<b>0.466</b>	<b>0.468</b>	0.398	<b>0.574</b>	<b>0.533</b>	<b>0.498</b>

**Table 4.** Ablation study of some modules in UCDDMI: (1) MI optimization based on the whole network (**basic**), (2) adding the Jaccard similarity (**+Ja**), and (3) using the proposed new MI maximization mechanism (**+Fg**). The *NMI*, *ACC*, and *F1-score* are used to evaluate the performance of community detection. The best results are recorded in bold.

Framework	Cora			Citeseer			Pubmed		
	NMI	ACC	F1	NMI	ACC	F1	NMI	ACC	F1
Basic	0.457	0.651	0.566	0.431	0.672	<b>0.655</b>	0.288	0.608	0.599
Basic + Ja	0.514	0.700	0.639	0.437	0.687	0.638	0.293	0.615	0.604
Basic + Ja + Fg (our)	<b>0.573</b>	<b>0.732</b>	<b>0.715</b>	<b>0.441</b>	<b>0.694</b>	0.649	<b>0.297</b>	<b>0.692</b>	<b>0.688</b>

attributes or network structure, our UCDDMI achieves an obvious improvement. It demonstrates that the usefulness of both kinds of information of networks in community detection, and the rationality of UCDDMI when simultaneously using such information to learn high-quality node representation. Moreover, as for methods that use both the network structure and attributes, we find UCDDMI consistently outperforms these methods. The main reason is that UCDDMI can better utilize the information of networks. In particular, ARVGE only captures the information of 2-hop neighboring nodes, while our UCDDMI can exploit the global information without being restricted by  $k$ -order neighbors.

Besides, UCDDMI achieves better performance compared with DGI which also considers the global information by utilizing the strategy of MI maximization. This is because DGI implements such maximization based on the whole graph, which ignores the finer substructures of networks and learns relatively coarse node representations. In contrast, UCDDMI deals with finer substructures by implementing the MI maximization based on more fine-grained subnetwork, which enables it to better learn node representations and improves the performance of community detection.



#### 4.4. Ablation study

Two components, the Jaccard similarity and new MI maximization mechanism play significant roles in helping our model the better learn node representations and detect high-quality community structures of networks. Thus, we carry out an ablation study experiment to evaluate the contributions of these two components. In particular, ‘**basic**’ means the model that implements MI optimization based on the whole network. ‘+Ja’ denotes adding the Jaccard similarity to the model, and ‘+Fg’ represents the adoption of the new MI maximization mechanism based on the fine-grained subnetworks.

As shown in table 4, we find that using the Jaccard similarity to evaluate the importance between nodes for guiding the aggregation clearly improves the performance of community detection. This reveals that it is necessary to adjust the weight of nodes in the aggregation of features according to their importance. Moreover, by comparing the penultimate row and the second row of table 4, we find that the proposed new MI maximization mechanism performs better than the existing MI maximization strategy. This demonstrates the effectiveness of maximizing the MI between local node representations and more fine-grained subnetwork representation.

#### 4.5. Parameters analysis

Next, we analyze the influence of two key parameters in our proposed method: the embedding dimension (controlled by  $d$ ), and the coefficient of Jaccard similarity (controlled by  $\alpha$ ). As shown in figures 3(a)–(c), the experimental performance is improved by increasing the dimension  $d$ . Moreover, through the study of figures 3(d)–(f), we observe that the experimental performance obtained for  $\alpha > 0$  are always better than for when  $\alpha = 0$ . Such phenomena further demonstrates the effectiveness of using the Jaccard similarity to guide the aggregation. Moreover, we find a slight decline in experimental performances when adjusting  $\alpha$  beyond a certain value. After a careful investigation, we found that increasing the value of  $\alpha$  magnifies the differences between neighboring nodes. When the difference is magnified to a certain extent, the information of some neighboring nodes will be ignored, thus leading to a performance decline. It indicates that our proposed aggregation strategy can effectively distinguish the importance of different neighboring nodes.

#### 4.6. Visualization

To better understand the effectiveness of our proposed UCDMI, we visualize the community detection results of UCDMI and DGI on Polbooks, respectively. Based on the comparison of results between UCDMI and DGI, we expect to demonstrate the effectiveness of the proposed aggregation strategy and the new MI maximization mechanism. The results of visualizations are presented in figure 4, where the ground-truth communities are denoted by different shapes, while the communities detected by experiments are represented by different colors.

As we can see from figure 4, UCDMI exhibits three communities in Polbooks network. It is worth noticing that there are two communities, colored by red and yellow, that are almost detected correctly by our UCDMI. In comparison, four nodes, i.e.  $v_5$ ,  $v_{50}$ ,  $v_{67}$ , and  $v_{85}$  have been divided incorrectly into communities by DGI. Based on some careful investigations, we find that these nodes reside in the edge of the communities (called bridge nodes). They often share more Jaccard similarity, such as  $v_{67}$ , to nodes in the community where they belong. Thus, our proposed UCDMI, which uses Jaccard similarity to guide the aggregation, can effectively utilize such characteristic and detect high-quality communities. Besides, we find that UCDMI can detect communities where nodes within the same communities are closely connected, which is consistent with the definition of the community structure. We can conclude that the proposed aggregation strategy and the new MI maximization mechanism are able to preserve the more fine-grained global information, e.g. the information of community structures, and are suitable for community detection.

### 5. Conclusion

In this paper, we propose a novel method based on MI maximization for community detection in attributed networks. To increase the contribution of more important nodes in the aggregation process, we design a new aggregation of GCN which can distinguish the importance between nodes by measuring the Jaccard similarity of nodes. To exploit the fine-grained global information of networks, we develop a novel MI maximization mechanism to maximize the MI between the local node representations and global subnetwork representations. Some experiments demonstrate the effectiveness of our proposed method in community detection and it achieve better results than state-of-the-art baseline methods.

### Acknowledgments

Dr. Xianghua Li and Professor Zhen Wang are the corresponding authors of this paper. This work was supported by the National Natural Science Foundation for Distinguished Young Scholars (Grant No. 62025602), National Natural Science Foundation of China (Grant Nos. 61976181, U1803263, 11931015 and 61762020), the Fok Ying-Tong Education Foundation, China (Grant No. 171105), the Technological Innovation Team of Shaanxi Province (No. 2020TD-013), the Science and Technology Support Program of Guizhou (Grant No. QKHZC2021YB531), and the Russian Ministry of Science and Education Agreement (Grant No. 13.1902.21.0026).

### Data availability statement

No new data were created or analyzed in this study.

## References

- [1] Wang Z, Wang C, Li X, Gao C, Li X and Zhu J 2020 *IEEE Trans. Knowl. Data Eng.* <https://doi.org/10.1109/TKDE.2020.2997043>
- [2] Li H-J, Wang L, Zhang Y and Perc M 2020 *New J. Phys.* **22** 063035
- [3] Li X, Chen M, Nie F and Wang Q 2017 multiview-based parameter free framework for group detection *Proc. 31st AAAI Conf. Artificial Intelligence* pp 4147–53
- [4] Yu L, Li P, Zhang J and Kurths J 2021 *New J. Phys.* **23** 033029
- [5] Gao C, Su Z, Liu J and Kurths J 2019 *Commun. ACM* **62** 61–7
- [6] Li Y, Sha C, Huang X and Zhang Y 2018 Community detection in attributed graphs: an embedding approach *Proc. 32nd AAAI Conf. Artificial Intelligence* pp 338–45
- [7] Deng Y, Wang J, Gao C, Li X, Wang Z and Li X 2021 *Physica A* **576** 126058
- [8] Gao C, Fan Y, Jiang S, Deng Y, Liu J and Li X 2021 *IEEE Trans. Intell. Transport. Syst.* <https://doi.org/10.1109/TITS.2021.3058185>
- [9] Alessandro M and Vittorio C C 2018 *New J. Phys.* **20** 063022
- [10] Gao C and Liu J 2017 *IEEE Trans. Syst. Man Cybern. Syst.* **47** 171–83
- [11] Jin D, Chen Z and He D 2015 Modeling with node degree preservation can accurately find communities *Proc. 29th AAAI Conf. Artificial Intelligence (AAAI)* pp 160–7
- [12] Li X, Wang Z, Gao C and Shi L 2017 *Appl. Math. Comput.* **310** 182–93
- [13] Jia Y, Zhang Q, Zhang W and Wang X 2019 CommunityGAN: community detection with generative adversarial nets *Proc. World Wide Web Conf.* pp 784–94
- [14] Li X, Chen Y and Xu J 2021 *Stat. Sci.* **36** 2–15
- [15] Chen M, Kuzmin K and Szymanski B K 2014 *IEEE Trans. Comput. Soc. Syst.* **1** 46–65
- [16] Liu X, Du Y, Jiang M and Zeng X 2020 *IEEE Trans. Comput. Soc. Syst.* **7** 437–49
- [17] Cavallari S, Zheng V W, Cai H, Chang K C C and Cambria E 2017 Learning community embedding with community detection and node embedding on graphs *Proc. 2017 ACM on Conf. Information and Knowledge Management* pp 377–86
- [18] Zou H, Duan Z, Guo X, Zhao S, Chen J, Zhang Y and Tang J 2021 *Inf. Sci.* **562** 385–97
- [19] Gao C, Wang Z, Li X, Zhang Z and Zeng W 2016 *PLoS One* **11** e0161755
- [20] Wang X and Davidson I 2010 Active spectral clustering *Proc. 10th IEEE Int. Conf. Data Mining (IEEE)* pp 561–8
- [21] Jin D, Liu Z, Li W, He D and Zhang W 2019 Graph convolutional networks meet Markov random fields: semi-supervised community detection in attribute networks *Proc. 33rd AAAI Conf. Artificial Intelligence* vol 33 pp 152–9
- [22] Velickovic P, Fedus W, Hamilton W L, Lio P, Bengio Y and Hjelm R D 2019 Deep graph infomax *Proc. 7th Int. Conf. Learning Representations* pp 1–17
- [23] Chiang W L, Liu X, Si S, Li Y, Bengio S and Hsieh C J 2019 Cluster-GCN: an efficient algorithm for training deep and large graph convolutional networks *Proc. 25th ACM SIGKDD Int. Conf. Knowledge Discovery & Data Mining* pp 257–66
- [24] Chen H, Xu Y, Huang F, Deng Z, Huang W, Wang S, He P and Li Z 2020 Label-aware graph convolutional networks *Proc. 29th ACM Int. Conf. Information & Knowledge Management* pp 1977–80
- [25] Wang C, Pan S, Long G, Jiang J and Zhang C 2019 Attributed graph clustering: a deep attentional embedding approach *Proc. 28th Int. Joint Conf. Artificial Intelligence* pp 3670–6
- [26] Li Y, Gu C and Kohli P 2019 Graph matching networks for learning the similarity of graph structured objects *Int. Conf. Machine Learning (PMLR)* pp 3835–45
- [27] Hjelm R D, Fedorov A, Lavoie-Marchildon A and Bengio Y 2019 Learning deep representations by mutual information estimation and maximization *Proc. 7th Int. Conf. Learning Representations* pp 1–24
- [28] Li Z, Liu J and Wu K 2017 *IEEE Trans. Cybern.* **48** 1963–76
- [29] Gao C, Liang M, Li X, Zhang Z, Wang Z and Zhou Z 2018 *IEEE/ACM Trans. Comput. Biol. Bioinf.* **15** 1916–28
- [30] Li S, Jiang L, Wu X, Han W, Zhao D and Wang Z 2021 *Appl. Math. Comput.* **401** 126012
- [31] Newman M E 2006 *Phys. Rev. E* **74** 036104
- [32] Cao S, Lu W and Xu Q 2015 GraRep: learning graph representations with global structural information *Proc. 24th ACM Int. Conf. Information and Knowledge Management* pp 891–900
- [33] Perozzi B, Al-Rfou R and Skiena S 2014 DeepWalk: online learning of social representations *Proc. 20th ACM SIGKDD Int. Conf. Knowledge Discovery and Data Mining* pp 701–10
- [34] Cao S, Lu W and Xu Q 2016 Deep neural networks for learning graph representations *Proc. 30th AAAI Conf. Artificial Intelligence* pp 1145–52
- [35] Jiang S, Li X, Chen X, Wang Z, Perc M and Gao C 2021 *Europhys. Lett.* **135** 18001
- [36] Qin M and Lei K 2021 *Inf. Sci.* **551** 146–67
- [37] Chinrungrueng C and Sequin C H 1995 *IEEE Trans. Neural Netw.* **6** 157–69
- [38] Yu D, Xu H, Chen C P, Bai W and Wang Z 2021 *IEEE Trans. Ind. Electron.* <https://doi.org/10.1109/TIE.2021.3080205>
- [39] Blondel V D, Guillaume J-L, Lambiotte R and Lefebvre E 2008 *J. Stat. Mech.* P10008
- [40] Guo T, Pan S, Zhu X and Zhang C 2019 *IEEE Trans. Knowl. Data Eng.* **31** 706–19
- [41] Tandon A, Albeshri A, Alhalabi W and Fortunato S 2019 *Phys. Rev. E* **99** 042301
- [42] Zhang B, Yu Z and Zhang W 2020 Community-centric graph convolutional network for unsupervised community detection *Proc. 29th Int. Joint Conf. Artificial Intelligence* pp 3515–21
- [43] Sun F, Qu M, Hoffmann J and Tang J 2019 vGraph: a generative model for joint community detection and node representation learning *Advances in Neural Information Processing Systems 32: Annual Conf. Neural Information Processing Systems* pp 512–22
- [44] Pan S, Hu R, Long G, Jiang J, Yao L and Zhang C 2018 Adversarially regularized graph autoencoder for graph embedding *Proc. 27th Int. Joint Conf. Artificial Intelligence* pp 2609–15
- [45] Zhang T, Xiong Y, Zhang J, Zhang Y, Jiao Y and Zhu Y 2020 CommDGI: community detection oriented deep graph infomax *Proc. 29th ACM Int. Conf. Information & Knowledge Management* pp 1843–52
- [46] Wu Z, Pan S, Chen F, Long G, Zhang C and Yu P S 2021 A comprehensive survey on graph neural networks *IEEE Trans. Neural Netw. Learn. Syst.* **32** 4–24
- [47] Elhadi H and Agam G 2013 Structure and attributes community detection: comparative analysis of composite, ensemble and selection methods *Proc. 7th Workshop on Social Network Mining and Analysis* pp 1–7
- [48] Rossi R and Ahmed N 2015 The network data repository with interactive graph analytics and visualization *Proc. 29th AAAI Conf. Artificial Intelligence* pp 4292–3