Multiple Topics Community Detection in Attributed Networks

Chaobo He
School of Computer Science, South China Normal University
Pazhou Lab
Guangzhou, China
hechaobo@m.scnu.edu.cn

Junwei Cheng
School of Computer Science, South China Normal University
Guangzhou, China
jung@m.scnu.edu.cn

Guohua Chen
School of Computer Science, South China Normal University
Pazhou Lab
Guangzhou, China
chengh@m.scnu.edu.cn

Yong Tang*
School of Computer Science, South China Normal University
Pazhou Lab
Guangzhou, China
ytang@m.scnu.edu.cn

ABSTRACT

Since existing methods are often not effective to detect communities with multiple topics in attributed networks, we propose a method named SSAGCN via Autoencoder-style self-supervised learning. SSAGCN firstly designs an adaptive graph convolutional network (AGCN), which is treated as the encoder for fusing topology information and attribute information automatically, and then utilizes a dual decoder to simultaneously reconstruct network topology and attributes. By further introducing the modularity maximization and the joint optimization strategies, SSAGCN can detect communities with multiple topics in an end-to-end manner. Experimental results show that SSAGCN outperforms state-of-the-art approaches, and also can be used to conduct topic analysis well.

CCS CONCEPTS

• **Computing methodologies → Neural networks**; **Learning latent representations**.

KEYWORDS

community detection, topic analysis, self-supervised learning, graph convolutional network, attributed networks

ACM Reference Format:
Chaobo He, Junwei Cheng, Guohua Chen, and Yong Tang. 2023. Multiple Topics Community Detection in Attributed Networks. In *Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval (SIGIR '23), July 23–27, 2023, Taipei, Taiwan.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3539618.3592026

*Corresponding author.

SIGIR '23, July 23–27, 2023, Taipei, Taiwan.

ACM ISBN 978-1-4503-9408-6/23/07…$15.00
https://doi.org/10.1145/3539618.3592026

1 INTRODUCTION

Community detection is an important task for analyzing attributed networks with both topology information and attribute information. It aims to identify nodes clusters with high cohesiveness. Namely, nodes in the same cluster link to each other more densely, and meanwhile share more similar attributes than those in other clusters [1, 2]. Detecting communities effectively is important for analyzing the organizational structures and understanding the functions of attributed networks. In recent years, some community detection methods for attributed networks have been proposed, and can be mainly divided into four categories according to the model framework used, including nonnegative matrix factorization (NMF)-based methods (e.g., SCI [3], ANEM [4] and AGNMF-AN [5]), probabilistic generative model-based methods (e.g., GBAGC [6], SCANNER [7], PAICAN [8] and AdaMRF [9]), game theory-based methods (e.g., DCFG [10] and DGTA [11]), and more promising graph neural networks (GNN)-based methods (e.g., MOEA [12], GUCD [13], SDCN [14], ComGA [15], PCCD [16] and other methods reviewed in [17]). In general, existing methods all have achieved different levels of performance, but they still suffer from the following two problems:

- **They are ubable to fuse topology information and attribute information adaptively.** It is generally believed that fusing topology information and attribute information facilitates better community partitioning, because these two types of information can interact with and complement each other. Due to the differences in terms of both function and quality, it is necessary to balance the contribution of every type of information. To this end, most existing methods often manually set the corresponding hyperparameters, which can not be learned automatically and accurately, resulting in the extra tuning cost and unstable performance.
- **They often ignore extracting the latent multiple topics associated with communities.** Topics are closely related to communities in attributed networks. Uncovering these latent topics can interpret communities from the semantic aspect and offer more insights into communities, but the vast majority of existing methods pay little attention to this. Although several pioneer methods (e.g., SA-Cluster [2] and SCI [3]) have made some efforts, they all treat the community

and the topic in a one-to-one relationship. However, in real-world attributed networks, one community can correspond to multiple topics and multiple communities can share the same topic. Namely, the relationship between them should be many-to-many.

To address these problems above, in this paper we propose a method for detecting communities with multiple topics using self-supervised adaptive graph convolutional networks (SSAGCN for short hereafter). Our main contributions are summarized as follows.

- We design an adaptive graph convolutional network (AGCN) for fusing topology information and attribute information. AGCN firstly performs graph convolutional operations with sharing parameters over topology graph and attribute similarity graph, and then introduces the attention mechanism to learn the contribution weights of these two types of information adaptively.
- We adopt the Autoencoder-style self-supervised learning to develop an end-to-end framework for SSAGCN. Specifically, AGCN driven by modularity maximization is treated as the encoder, and a dual decoder is used for reconstructing network topology and attributes. The results of extensive experiments show the superiority of SSAGCN.

2 METHODOLOGY

2.1 Problem Formulation

Without loss of generality, a given attributed network is modeled as an undirected and unweighted graph $G = \{V, E, \mathbf{A}, \mathbf{X}\}$, where $V = \{v_1, v_2, ..., v_n\}$ is the set of n nodes, $E = \{e_{ij} | v_i \in V \wedge v_j \in V\}$ is the set of m edges between nodes, $\mathbf{A} = [\mathbf{A}_{ij}]^{n\times n}$ is the adjacency matrix and $\mathbf{X} = [\mathbf{X}_{iq}]^{n\times l}$ is the node-attribute matrix. If $e_{ij} \in E$, $\mathbf{A}_{ij} = 1$, and $\mathbf{A}_{ij} = 0$ otherwise. For the set of l attributes $Y = \{y_1, y_2, ..., y_l\}$, if v_i has attribute y_q, $\mathbf{X}_{iq} = 1$, and $\mathbf{X}_{iq} = 0$ otherwise. $\mathbf{A}$ and $\mathbf{X}$ are utilized to represent the topology information and the attribute information of G, respectively.

Given the definitions above, detecting communities with multiple topics aims to divide nodes in G into k disjoint clusters $C = \{c_1, c_2, ..., c_k\}$ with d topics $Z = \{z_1, z_2, ..., z_d\}$, so that (1) nodes within the same cluster connect to each other more densely and have more similar attribute values than those outside, and (2) every community c_i is associated with a topic distribution vector $\mathbf{S}_i = [\mathbf{S}_{i1}, \mathbf{S}_{i2}, ..., \mathbf{S}_{id}]$, where $\mathbf{S}_{ij}$ denotes the probability that the topic z_j belongs to c_i, and every topic can be represented as the distribution over l-dimensional attribute space.

2.2 SSAGCN Method

1) Overview. To fuse topology information and attribute information for community detection, we firstly treat G as the topology graph $G_t = \{V, E_t, \mathbf{A}_t, \mathbf{X}\}$ ($E_t = E$ and $\mathbf{A}_t = \mathbf{A}$), and construct a K-nearest neighbor graph as the attribute similarity graph $G_a = \{V, E_a, \mathbf{A}_a, \mathbf{X}\}$ by calculating cosine similarities between node attribute vectors in $\mathbf{X}$, where E_a and $\mathbf{A}_a$ respectively denote the edges set and the binary-valued adjacency matrix of G_a. Then we apply two GCNs with shared parameters to respectively learn node representations $\mathbf{H}_t$ and $\mathbf{H}_a$ from G_t and G_a. Due to the coexistence of homogeneity and heterogeneity between G_t and G_a, $\mathbf{H}_t$ and $\mathbf{H}_a$ will contain the consensus and differential features, and they both have different functions for community detection. In view of this, we introduce the attention mechanism to adaptively incorporate $\mathbf{H}_t$ and $\mathbf{H}_a$ into the final node representation $\mathbf{H}$. For driving $\mathbf{H}$ to represent community memberships and extracting community topics simultaneously, we uniformly set the dimensions of $\mathbf{H}_t$, $\mathbf{H}_a$ and $\mathbf{H}$ to $n\times k$, and train two GCNs and the attention network (collectively referred to as Adaptive GCN or AGCN) by the unified objective of modularity maximization followed by reconstructing network topology and attributes.

Following the idea above, we adopts Autoencoder-style self-supervised learning to design SSAGCN, which is mainly composed of three modules: AGCN encoder, Modularity maximization and Dual decoder. Next, we will detail every module.

2) AGCN encoder. AGCN encoder is composed of two GCNs with shared parameters. One GCN is used to learn node representation from the topology graph G_t as follows:

$$\mathbf{H}_t^{(L)} = \mathrm{Relu}(\widetilde{\mathbf{D}_t}^{-\frac{1}{2}} \widetilde{\mathbf{A}}_t \widetilde{\mathbf{D}_t}^{-\frac{1}{2}} \mathbf{H}_t^{(L-1)} \mathbf{W}^{(L)}), \tag{1}$$

where $\mathbf{W}^{(L)}$ is the weight matrix of the L-th layer in GCN, $\widetilde{\mathbf{A}}_t = \mathbf{A}_t + \mathbf{I}$, $\widetilde{\mathbf{D}}_t$ is the diagonal degree matrix of $\widetilde{\mathbf{A}}_t$ and Relu is the activation function. We set $\mathbf{H}_t^{(0)} = \mathbf{X}$ and denote the last layer output representation as $\mathbf{H}_t$. Using the same architecture, the other GCN sharing $\mathbf{W}^{(L)}$ is used to learn node representation $\mathbf{H}_a$ from the attribute similarity graph G_a. To fuse $\mathbf{H}_t$ and $\mathbf{H}_a$ adaptively for community detection, we introduce the attention mechanism to learn their corresponding importance: $(\delta_t, \delta_a) = \mathrm{Attention}(\mathbf{H}_t, \mathbf{H}_a)$, where δ_t, $\delta_a \in \mathbb{R}^{n\times 1}$. For $\mathbf{H}_t$, we firstly transform it through the LeakyRelu function, and then use one shared attention matrix $\mathbf{\Theta} \in \mathbb{R}^{1\times k'}$ to learn its attention value $\omega_t \in \mathbb{R}^{n\times 1}$ as follows:

$$\omega_t = \mathrm{LeakyRelu}(\mathbf{H}_t \mathbf{W}_t^T + \mathbf{B}_t)\mathbf{\Theta}^T, \tag{2}$$

where $\mathbf{W}_t \in \mathbb{R}^{k'\times k}$ and $\mathbf{B}_t \in \mathbb{R}^{n\times k'}$ are respectively the weight and bias matrices. Following the same way, we can also obtain the attention value $\omega_a \in \mathbb{R}^{n\times 1}$ of $\mathbf{H}_a$. And then, we normalize ω_t to get δ_t via the softmax function: $\delta_t = \mathrm{softmax}(\omega_t) = \frac{\exp(\omega_t)}{\exp(\omega_a)+\exp(\omega_t)}$. Similarly, $\delta_a = \mathrm{softmax}(\omega_a)$. Let $\boldsymbol{\delta}_t = \mathrm{diag}(\delta_t) \in \mathbb{R}^{n\times n}$ and $\boldsymbol{\delta}_a = \mathrm{diag}(\delta_a) \in \mathbb{R}^{n\times n}$. The final node representation $\mathbf{H}$ is the sum of weighted $\mathbf{H}_t$ and $\mathbf{H}_a$: $\mathbf{H} = \boldsymbol{\delta}_t \mathbf{H}_t + \boldsymbol{\delta}_a \mathbf{H}_a$.

3) Modularity maximization module. To represent community membership better, we introduce modularity maximization metric [18], which has been widely used to model the community structure, to refine $\mathbf{H}$. Specifically, if we treat $\mathbf{H}$ as the community memberships matrix, the modularity Q of G can be approximately defined as $Q = \frac{1}{4|E|}\sum_{i,j}^{n}(\mathbf{A}_{ij} - \frac{d_i d_j}{2|E|})\mathbf{H}_i\mathbf{H}_j^T$, where d_i is the degree of v_i. Let $\mathbf{M} = [\mathbf{M}_{ij}]^{n\times n}$ whose element $\mathbf{M}_{ij} = \mathbf{A}_{ij} - \frac{d_i d_j}{2|E|}$, and omitting the constant $\frac{1}{4|E|}$ that has no effect on the maximum of Q. We denote Q as the following trace form:

$$Q = tr(\mathbf{H}^T\mathbf{M}\mathbf{H}) \quad s.t., \sum_{j=1}^{k} \mathbf{H}_{ij} = 1. \tag{3}$$

We take $-Q$ as the loss $\mathcal{L}_{\mathrm{MM}}$ to train AGCN, which can drive it to output discriminative community membership representation $\mathbf{H}$.

4) Dual decoder. The dual decoder is composed of two decoders, which are used for reconstructing network topology and attributes, respectively. The decoder for reconstructing network topology is inspired by the idea of community detection model using nonnegative matrix tri-factorization: $\mathbf{A} \approx \mathbf{H}\Omega\mathbf{H}^T$ [19], where Ω denotes the community interaction matrix with nonnegative constraint. We use this model to reconstruct the network topology $\hat{\mathbf{A}} = [\hat{\mathbf{A}}_{ij}]^{n\times n}$, and treat it as a layer of this decoder neural network by further imposing a sigmoid activation function as:

$$\hat{\mathbf{A}} = \mathrm{sigmoid}(\mathbf{H}\Omega\mathbf{H}^T) \quad s.t., \Omega \geq 0, \tag{4}$$

where Ω becomes the learnable weight parameter. Letting $\hat{\mathbf{A}}$ approximate to $\mathbf{A}$, we apply the following binary cross entropy function to measure the loss of reconstructing network topology:

$$\mathcal{L}_{\mathrm{Rec\_T}} = -\frac{1}{n^2}\sum_{i,j}^{n}(\mathbf{A}_{ij}\log\hat{\mathbf{A}}_{ij} + (1-\mathbf{A}_{ij})\log(1-\hat{\mathbf{A}}_{ij})). \tag{5}$$

To extract communities with multiple topics, we respectively represent every node, every community and every topic as the probability distribution of k communities, the distribution of d topics and the distribution of l attributes. Supposing that every node-attribute pair (v_i, y_q) is generated independently, the probability that v_i has the attribute y_q can be defined as:

$$\hat{\mathbf{X}}_{iq} = P(y_q|v_i) = \sum_{z_p\in Z}\sum_{c_j\in C} P(y_q|z_p)P(z_p|c_j)P(c_j|v_i), \tag{6}$$

where $P(c_j|v_i)$ denotes the probability that v_i belongs to community c_j, $P(z_p|c_j)$ denotes the probability that c_j is associated to topic z_p, and $P(y_q|z_p)$ denotes the probability that z_p has attribute y_q. Let $P(c_j|v_i) = \mathbf{H}_{ij}$, $\hat{\mathbf{X}} = [\hat{\mathbf{X}}_{iq}]^{n\times l}$, $\mathbf{S} = [\mathbf{S}_{jp}]^{k\times d}$ be the community-topic matrix and $\mathbf{U} = [\mathbf{U}_{pq}]^{d\times l}$ be the topic-attribute matrix. Eq. (6) can be rewritten as the matrix form:

$$\hat{\mathbf{X}} = \mathrm{sigmoid}(\mathbf{HSU}) \quad s.t., \sum_{p=1}^{d}\mathbf{S}_{jp} = 1, \sum_{q=1}^{l}\mathbf{U}_{pq} = 1. \tag{7}$$

We take the above model as two layers of a neural network served as the decoder for reconstructing network attributes, where both $\mathbf{S}$ and $\mathbf{U}$ become the trainable weight parameters. Furthermore, by this model the probability generated the attribute matrix $\mathbf{X}$ can be denoted as $P(\mathbf{X}|V) = \prod_{i=1}^{n}\prod_{q=1}^{l}(\hat{\mathbf{X}}_{iq})^{\mathbf{X}_{iq}}$. The objective of this attributes generative model is to maximize this likelihood, so we can adopt its negative log-likelihood of $P(\mathbf{X}|V)$ as the loss of reconstructing network attributes:

$$\mathcal{L}_{\mathrm{Rec\_A}} = -\sum_{i=1}^{n}\sum_{q=1}^{l}\mathbf{X}_{iq}\log\hat{\mathbf{X}}_{iq}. \tag{8}$$

5) The unified objective. Combing the losses of all modules above, we have the following unified objective function:

$$\mathcal{L} = \mathcal{L}_{\mathrm{Rec\_T}} + \mathcal{L}_{\mathrm{Rec\_A}} + \alpha\mathcal{L}_{\mathrm{MM}}, \tag{9}$$

where α is the trade-off parameter used to control the contribution of the modularity maximization module. Note that here we do not set the weights for $\mathcal{L}_{\mathrm{Rec\_T}}$ and $\mathcal{L}_{\mathrm{Rec\_A}}$, because we have introduced the adaptive information fusion mechanism in AGCN encoder, which can automatically learn the weights of topology information and attribute information. We use $\mathcal{L}$ to train AGCN, and when the training process converge the community index r of v_i can be determined by: $r = \operatorname{argmax}_{j=1...k} \mathbf{H}_{ij}$. Besides, for every community its affiliated topics and the corresponding key attributes can be extracted from $\mathbf{S}$ and $\mathbf{U}$, respectively.

3 EXPERIMENTS

3.1 Experiment Setup

1) Datasets. We select five widely used attributed networks with ground-truth community labels as datasets, including three citation networks: Cora, Citeseer and Pubmed [20], and two social networks: Blogcatalog and Flickr [21]. Every node in these networks is associated with a 0/1-valued attribute vector. The detail statistics of selected datasets are given in Table 1.

Table 1: Statistics of datasets.

Datasets	Type	n	m	l	k
Cora	Citation	2,708	5,429	1,433	7
Citeseer	Citation	3,312	4,732	3,703	6
Pubmed	Citation	19,729	44,338	500	3
Blogcatalog	Social	5,196	171,743	8,189	6
Flickr	Social	7,575	239,738	12,047	9

2) Baselines. As mentioned in Section 1, community detection methods that can integrate topology information and attribute information often perform better than those using only one type of information. Therefore, here we only select methods which can simultaneously utilize two types of information as baselines. These baselines include GNN-based methods GUCD [13], SDCN [14] and DAEGC [22], probabilistic generative model-based method AdaMRF [9], game theory-based method DGTA [11], NMF-based methods SCI [3] and ANEM [4].

3) Evaluation metrics. To evaluate SSAGCN, we select two widely used metrics: Normalized Mutual Information (NMI) [23] and Adjusted Rand Index (ARI) [23], where NMI $\in [0, 1]$ and ARI $\in [0, 1]$. Larger NMI and ARI scores indicate better performance.

4) Parameter settings. For SSAGCN, on all datasets its architecture of AGCN encoder is l-128-k. For the convenience of analysis, we set the topics number d to k (i.e., the numbers of the ground-truth community labels). We set $\alpha = 1$, the number of the nearest neighbors (i.e., K) used to construct the attribute similarity graph is set to be: 10 on Cora and Citeseer, 15 on Pubmed and 30 on Blogcatalog and Flickr. Due to the limitation of space, here we omit the deep analysis of the settings of K and α. As for all baselines, we retain the settings suggested in the corresponding papers. To ensure the fairness, we run all the methods 10 times and report the average performance.

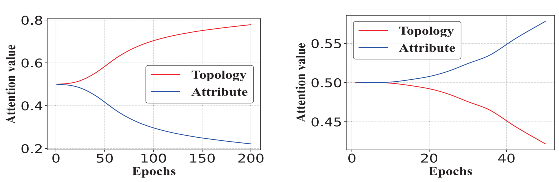
3.2 Results and Analysis

1) Comparison with baselines. We compare SSAGCN with baselines on every dataset and the results are presented in Table 2. As we can see, SSAGCN performs the best on Cora, Pubmed, Blogcatalog and Flickr networks, and on Citeseer also achieves the second-best performance in both NMI and ARI. Specially, SSAGCN has more obvious advantages on Blogcatalog and Flickr networks. On Blogcatalog, compared with the second best results, SSAGCN achieves 26.4% and 12.1%, improvements in terms of NMI and ARI, respectively, and on Flickr the improvement rates are respectively 12.3% and 19.1%. These results demonstrate the effectiveness and superiority of SSAGCN. This benefits from its features of adaptive information fusion and detecting communities with multiple topics, which all baseline methods do not have.

Table 2: Comparison results with baselines. Bold indicates the best, underlined indicates the second best.

Datasets	Metrics	GUCD	SDCN	DAEGC	AdaMRF	DGTA	SCI	ANEM	SSAGCN
Cora	NMI	0.323	0.491	0.528	0.362	0.451	0.382	0.345	**0.565**
	ARI	0.319	0.413	0.496	0.304	0.431	0.343	0.304	**0.515**
Citeseer	NMI	0.274	0.387	**0.397**	0.288	0.217	0.383	0.281	0.391
	ARI	0.232	**0.402**	0.378	0.273	0.119	0.312	0.262	0.379
Pubmed	NMI	0.269	0.253	0.243	0.211	0.231	0.261	0.234	**0.327**
	ARI	0.211	0.202	0.217	0.209	0.205	0.201	0.204	**0.288**
Blogcatalog	NMI	0.302	0.311	0.277	0.289	0.256	0.302	0.308	**0.393**
	ARI	0.231	0.216	0.202	0.223	0.221	0.213	0.212	**0.259**
Flickr	NMI	0.271	0.359	0.308	0.253	0.239	0.357	0.313	**0.403**
	ARI	0.209	0.223	0.215	0.206	0.227	0.231	0.211	**0.275**

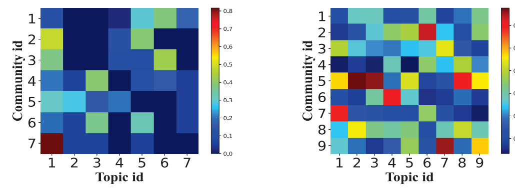
2) Adaptive information fusion analysis. In SSAGCN, we introduce the attention mechanism to adaptively learn the weights of topology information (encoded as $\mathbf{H}_t$) and attribute information (encoded as $\mathbf{H}_a$). To verify its effectiveness, we take Cora and Flickr as examples and present the changing trends of average attention values during the training process (Figure 1). It can be seen that the attention value of topology information and attribute information are almost the same at the beginning, but become different when the training epoch increases. On Cora the attention value of topology information gradually increases, while the attention value of attribute information keeps decreasing, but it is opposite on Flickr. These findings fully demonstrate that the attention mechanism used here can help SSAGCN learn the importance of different types of information step by step.



(a) Cora (b) Flickr

Figure 1: The attention changing trends w.r.t. epochs.

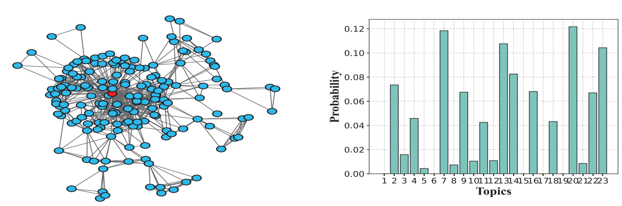
3) Detecting communities with multiple topics. We also take Cora and Flickr as examples, and visualize their community-topic matrices (i.e., S) on Figure 2. It can be clearly seen that many communities are associated with multiple topics. For example, on Cora c_1 has strong correlations with both t_5 and t_6, and on Flickr c_5 has strong correlations with both t_2 and t_3. In real-world, it is common that papers (e.g., nodes in Cora, Citeseer and Pubmed) in the same community (cluster) share multiple similar research topics. Similarly, users in social networks (e.g., nodes in Blogcatalog and Flickr) often have multiple common interests. These visualization results well demonstrate these phenomenons.



(a) Cora (b) Flickr

Figure 2: Community distribution on topics.

4) A case study on SIGIR co-authorship network. To further demonstrate this topic analysis ability of SSAGCN, we conduct a case study on SIGIR co-authorship network. This network is extracted from SIGIR papers published between 2013 and 2022, and is composed of 5830 nodes, 48252 links and 3901 attributes. We first use Louvain algorithm [24] to determine the best number of communities: 451, and then set $t = 23$ according to the perplexity metric used in LDA topic model [25]. As we known, a famous scholar usually leads a big team with multiple research directions. This means that his/her ego-centric community will share multiple topics. In view of this, we take a famous IR scholar "Maarten de Rijke" as an example, and perform the relevant analysis. His community networks (the red node denotes Maarten de Rijke) and community-topic distributions are visualized in Figure 3. As we can see, Maarten de Rijke's community has dense internal links and has multiple prominent topics. For example, it is obviously associated with topic t_7 and t_{20}.



(a) Community network (b) Topic distribution

Figure 3: Maarten de Rijke's community.

4 CONCLUSIONS AND FUTURE WORK

To detect communities with multiple topics in attributed networks, in this paper we propose a method named SSAGCN, which is based on GNN and self-supervised learning. SSAGCN has two significant features: adaptive fusion of topology information and attribute information, and end-to-end learning of communities with multiple topics. Extensive experiments on several real attributed networks demonstrate its superiority over state-of-the-art approaches. In the future, we will extend it to dynamic attributed networks, and trace the evolutions of communities and topics.

ACKNOWLEDGMENTS

This work was supported by the National Natural Science Foundation of China under Grant 62077045.

REFERENCES

[1] Petr Chunaev. Community detection in node-attributed social networks: a survey. 2020. *Computer Science Review* 37 (2020), 100286.

[2] Hong Cheng, Yang Zhou, and Jeffrey Xu Yu. 2011. Clustering large attributed graphs: a balance between structural and attribute similarities. *ACM Transactions on Knowledge Discovery from Data* 5,2 (2011), 12.

[3] Xiao Wang, Di Jin, Xiaochun Cao, Liang Yang, and Weixiong Zhang. 2016. Semantic community identification in large attribute networks. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence*. 265–271.

[4] Juanhui Li, Ling Huang, Changdong Wang, Dong Huang, Jianhuang Lai, and Pei Chen. 2021. Attributed network embedding with micro-meso structure. *ACM Transactions on Knowledge Discovery from Data* 15, 4 (2021), 72.

[5] Kamal Berahmand, Mehrnoush Mohammadi, Farid Saberi-Movahed, Yuefeng Li, and Yue Xu. 2023. Graph regularized nonnegative matrix factorization for community detection in attributed networks. *IEEE Transactions on Network Science and Engineering* 10, 1 (2023), 372-385.

[6] Zhiqiang Xu, Yiping Ke, Yi Wang, Hong Cheng, and James Cheng. 2014. GBAGC: a general Bayesian framework for attributed graph clustering. *ACM Transactions on Knowledge Discovery from Data* 9, 1 (2014), 5.

[7] Gianni Costa and Riccardo Ortale. 2022. Overlapping communities and roles in networks with node attributes: probabilistic graphical modeling, Bayesian formulation and variational inference. *Artificial Intelligence* 302 (2022), 103580.

[8] Aleksandar Bojchevski and Stephan Günnemann. Bayesian robust attributed graph clustering: joint learning of partial anomalies and group structure. 2018. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence*. 2738-2745.

[9] Dongxiao He, Huixin Liu, Zhiyong Feng, Xiaobao Wang, Di Jin, Wenze Song, and Yuxiao Huang. 2022. A joint community detection model: integrating directed and undirected probabilistic graphical models via factor graph with attention mechanism. *IEEE Transactions on Big Data* 8, 4 (2022), 994-1006.

[10] Zhan Bu, Huijia Li, Jie Cao, Zhen Wang, and Guangliang Gao. 2019. Dynamic cluster formation game for attributed graph clustering. *IEEE Transactions on Cybernetics* 49,1 (2019), 328-341.

[11] Lu Chen, Yunjun Gao, Xingrui Huang, Christian S. Jensen, and Bolong Zheng. 2022. Efficient distributed clustering algorithms on star-schema heterogeneous graphs. *IEEE Transactions on Knowledge and Data Engineering* 34, 10 (2022), 4781-4796.

[12] Jianyong Sun, Wei Zheng, Qingfu Zhang, Zongben Xu. 2022. Graph neural network encoding for community detection in attribute networks. *IEEE Transactions on Cybernetics* 52, 8 (2022), 7791-7804.

[13] Dongxiao He, Yue Song, Di Jin, Zhiyong Feng, Binbin Zhang, Zhizhi Yu, and Weixiong Zhang. 2020. Community-centric graph convolutional network for unsupervised community detection. In *Proceedings of the 29th International Joint Conference on Artificial Intelligence*. 3515-3521.

[14] Deyu Bo, Xiao Wang, Chuan Shi, Meiqi Zhu, Emiao Lu, and Peng Cui. Structural deep clustering network. 2020. In *Proceedings of The 2020 Web Conference*. 1400-1410.

[15] Xuexiong Luo, Jia Wu, Amin Beheshti, Jian Yang, Xiankun Zhang, Yuan Wang, Shan Xue. 2022. ComGA: community-aware attributed graph anomaly detection. In *Proceedings of the 15th ACM International Conference on Web Search and Data Mining*. 657-665.

[16] Zheng Gao, Hongsong Li, Zhuoren Jiang, and Xiaozhong Liu. Detecting user community in sparse domain via cross-graph pairwise learning. 2020. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 139–148.

[17] Di Jin, Zhizhi Yu, Pengfei Jiao, Shirui Pan, Dongxiao He, Jia Wu, Philip S. Yu, and Weixiong Zhang. 2023. A survey of community detection approaches: from statistical modeling to deep learning. *IEEE Transactions on Knowledge and Data Engineering* 35 (2) (2023) 1149-1170.

[18] M.E.J. Newman. Modularity and community structure in networks. 2006. *Proceedings of the national academy of sciences* 3 (23) (2006) 8577-8582.

[19] Chaobo He, Xiang Fei, Qiwei Cheng, Hanchao Li, Zeng Hu, and Yong Tang. 2022. A survey of community detection in complex networks using nonnegative matrix factorization. *IEEE Transactions on Computational Social Systems* 9, 2 (2022), 440-457.

[20] Prithviraj Sen, Galileo Namata, Mustafa Bilgic, Lise Getoor, Brian Galligher, and Tina Eliassi-Rad. 2008. Collective classification in network data. *AI Magazine* 29, 3 (2008), 93-106.

[21] Jundong Li, Xia Hu, Jiliang Tang, and Huan Liu. Unsupervised streaming feature selection in social media. 2015. In *Proceedings of the 24th International Conference on Information and Knowledge Management*. 1041-1050.

[22] Chun Wang, Shirui Pan, Ruiqi Hu, Guodong Long, Jing Jiang, and Chengqi Zhang. Attributed graph clustering: a deep attentional embedding approach. 2019. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*. 3670-3676.

[23] Tanmoy Chakraborty, Ayushi Dalmia, Animesh Mukherjee, and Niloy Ganguly. 2018. Metrics for community analysis: a survey. *ACM Computing Survey* 50, 4 (2018), 54.

[24] Vincent D Blondel, Jean-Loup Guillaume, Renaud Lambiotte, and Etienne Lefebvre. 2008. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment* 10 (2008), P10008.

[25] David Meir Blei, Andrew Ng, and MI Jordan. Latent dirichlet allocation. 2003. *The Journal of Machine Learning Research* 3 (2003), 993-1022.