Information Fusion 105 (2024) 102254

Contents lists available at ScienceDirect

Information Fusion

journal homepage: www.elsevier.com/locate/inffus

Full length article

Detecting communities with multiple topics in attributed networks via self-supervised adaptive graph convolutional network

Chaobo He [a,b], Junwei Cheng [a], Guohua Chen [a,b], Quanlong Guan [c,e], Xiang Fei [d], Yong Tang [a,b,*]

[a] School of Computer Science, South China Normal University, Guangzhou, China
[b] Pazhou Lab, Guangzhou, China
[c] College of Information Science and Technology, Jinan University, Guangzhou, China
[d] Department of Computing, Coventry University, Coventry, UK
[e] Guangdong Institution of Smart Education, Jinan University, Guangzhou, China

ARTICLE INFO

Keywords:
Community detection
Multiple topics community
Self-supervised learning
Graph convolutional network
Attributed networks

ABSTRACT

Attributed networks which have topology information and attribute information simultaneously are very common in the real world, e.g., online social networks and co-authorship networks. Community detection in attribute networks is a very valuable research topic. Although there is already a lot of work focused on it, but most of existing approaches still face two key challenges: topology information and attribute information fusion, and multiple topics community detection. To effectively address these challenges, in this paper we devise an approach called SSAGCN, which adopts the self-supervised learning paradigm using the autoencoder architecture. Specifically, SSAGCN comprises three main parts: adaptive graph convolutional network (AGCN) encoder, modularity maximization and dual decoder. AGCN uses two GCNs with shared parameters and the attention mechanism to fuse topology information and attribute information automatically. To drive AGCN encoder to uncover community structure, we select the modularity maximization as the optimization objective. The dual decoder is applied to reconstruct both topology structure and network attributes. By introducing the joint training strategy, SSAGCN is able to discover multiple topics community through the end-to-end manner. Extensive experiments are conducted on nine benchmark attributed networks, and the results illustrate not only the superiority of SSAGCN over state-of-the-art approaches, but also its good ability of community-topic analysis. For the reproducibility, we release the source code at https://github.com/GDM-SCNU/SSAGCN.

1. Introduction

In the current era of big data, many real-world complex networks no longer only have topology information, but also contain abundant attribute information. For example, in online social networks, users are often annotated with tags extracted from their profiles, posts and comments. Authors in co-authorship networks are associated with labels stemmed from their publications information, such as titles, abstracts and keywords. Formally, these networks with attribute information are called attributed networks. Due to the feature that they are more expressive and informative than networks with topology information only, attributed networks are more valuable and have attracted increasing attention from both industry and academia.

Community detection is a popular topic in the field of attributed networks analysis. It is essentially a node clustering problem that community and cluster have similar meaning. Although there is no unified definition yet, it is widely believed that nodes in the same community have more dense links and common attributes than those in different communities. Effective community detection is invaluable, because it can be used to analyze the structure patterns and function modules hidden in attributed networks [1,2]. In the last decade, for attributed networks various community detection approaches have been continuously presented. For example, SCI [3], CDE [4] and ANEM [5] methods use the joint Nonnegative Matrix Factorization (NMF) to integrate topology information and attribute information, and can directly identify community structure by means of the inherent clustering ability of NMF. GBAGC [6], AdaMRF [7] and PAICAN [8] apply the probabilistic generative framework with multiple observed variables to combine topology information and attribute information, and is able to learn the community distribution by inferring the corresponding latent variables. Following the game theory, DGTA [9] and DCFG [10] treat community detection in attributed networks as a dynamic cluster formation game,

* Corresponding author.
E-mail addresses: hechaobo@m.scnu.edu.cn (C. He), jung@m.scnu.edu.cn (J. Cheng), chengh@m.scnu.edu.cn (G. Chen), gql@jnu.edu.cn (Q. Guan), aa5861@coventry.ac.uk (X. Fei), ytang@m.scnu.edu.cn (Y. Tang).

https://doi.org/10.1016/j.inffus.2024.102254
Received 26 February 2023; Received in revised form 10 September 2023; Accepted 10 January 2024
Available online 13 January 2024
1566-2535/© 2024 Elsevier B.V. All rights reserved.

and using the benefit maximization strategy to determine the right community of every node. As the current promising methods, MOEA [11], SDCN [12] and GUCD [13] utilize graph neural networks (GNN) to fuse topology information and attribute information, and use clustering algorithms or the end-to-end learning way to detect communities. More GNN-based methods are comprehensively reviewed in [14]. Generally, these existing approaches have shown different degrees of effectiveness, but they still encounter two significant challenges as follows:

- **Adaptive fusion of topology information and attribute information.** Generally, topology information and attribute information fusion can lead to better community detection results. This is attributed to the fact that topology information and attribute information usually can compensate for each other, which facilitates learning more discriminative node representations. Considering that these two types of information often have differences in the quality and function, it is very necessary to balance their respective contributions (weights) in the task of community detection. After all, the effect of each type of information on the community memberships is different. In view of this, most existing approaches, especially NMF-based approaches (e.g., SCI, CDE and ANEM) often manually set the balance hyperparameter, which results in the additional tuning cost and the unstable performance. Thanks to the inherent mechanism of feature propagation and aggregation, GNN-based methods have the ability of information fusion. However, most of them are easily subjected to noise disturbance. For example, when adding a certain amount of random links into the attributed network or disrupting the attribute information, their performance will degrease significantly, which has been empirically studied in [15]. Therefore, GNN-based methods still need to become more robust via introducing the adaptive strategy, which is able to automatically and accurately learn the weights of topology information and attribute information. This adaptive strategy is beneficial to ensuring the good performance as much as possible, especially when topology information and attribute information have different qualities.
- **Detection of multiple topics communities.** In attributed networks, it is ubiquitous that communities are closely associated with topics. For example, in online social networks, users in the same community often share common topics of interest (e.g., Politics, Sports and Music), which are naturally identified by highly correlated attributes extracted from user generated content. In co-authorship networks, authors in the same community (research group) have common research topics (e.g., Data mining, Information retrieval and Machine learning), which can be inferred from keywords appeared in their publications. These underlying associated topics can offer the semantic interpretations and deeper insights to communities. In general, existing approaches almost ignore the problem of extracting topics from communities. Although a few pioneer approaches, such as SCI and SA-Cluster [2], have made some attempts, they all establish a one-to-one relationship between the community and topic. However, in real-world attributed networks (e.g., online social networks and co-authorship networks), a single community might associate with multiple topics, and multiple communities might have a common topic. In other words, the relationship between the community and the topic should be many-to-many. In view of this, multiple topics communities detection still needs further exploration.

Aiming at addressing these challenges above effectively, in this paper we incorporate community detection and topic extraction into a unified framework, and devise a method named SSAGCN for multiple topics community detection in attributed networks. In SSAGCN, we use Adaptive Graph Convolutional Networks (AGCN) to overcome these challenges of information fusion and detect communities with multiple topics via the self-supervised learning paradigm. So far as we know, it is the first time to develop a GCN-based approach for multiple topics community detection, and our main contributions include:

- To fuse topology information and attribute information, we devise an adaptive graph convolutional network (AGCN). AGCN is composed of two GCNs with shared parameters used to respectively learn node representations from the topology graph and attribute similarity graph. Through further introducing an attention mechanism, the contribution weights of different types of information can be learned adaptively.
- Following the idea of self-supervised learning, we design an end-to-end learning framework for SSAGCN by using the autoencoder architecture. Specifically, AGCN guided by the modularity maximization objective is regarded as the encoder, and utilizing a dual decoder to reconstruct both topology structure and network attributes. By training AGCN via the unified loss function, the representations of community-membership, community-topic and topic-attribute can be learned simultaneously.
- We conduct extensive experiments on 9 benchmark attributed networks. The results not only demonstrate the superior performance of SSAGCN over 10 highly competitive baselines, but also illustrate its excellent capability to semantically describe the communities using multiple topics.

The rest of this paper is organized as follows: Section 2 provides a concise overview of related work. We detail the proposed method SSAGCN in Section 3, and present comprehensive experimental results and analysis in Section 4. The conclusions are drawn in Section 5. Note that we have presented some parts of this paper in a short paper of SIGIR (Special Interest Group on Information Retrieval) conference, and made sufficient extensions here, especially the experimental part. Please refer to [16] for more details.

2. Related work

In this section, we outline some representative work on community detection in attributed networks and graph self-supervised learning, which are the most relevant to the topic focused on this paper.

2.1. Community detection in attributed networks

In the past decades, most community detection approaches mainly deal with complex networks with topology information only. However, with the emergence of various attributed networks (e.g., online social networks and co-authorship networks mentioned above), lots of community detection methods for attributed networks have been presented continuously. For example, SA-Cluster firstly constructs the attribute augmented graph, then designs a unified distance measure based on the neighborhood random walk to combine structural and attribute similarities, and finally applies k-means algorithm to detect communities. SCI, ANEM and AGNMF-AN [17] all utilize the NMF framework to jointly factorize topology structure matrix (e.g., adjacent matrix) and node-attribute matrix, and meanwhile introduce the consensus community membership matrix to make different types of information compensate for each other. From the generative perspective, AdaMRF utilizes Bayesian probabilistic model to infer communities, where links and attributes are treated as the observed variables, and communities are regarded as the hidden variables. Through the probabilistic inference process, topology information and attribute information can be fused naturally. Similar idea is adopted by GBAGC, PAICAN and SCANNER [18]. As one of the representative game theory-based methods, DGTA represents attributed networks as star-schema heterogeneous graphs modeling attributes as different types of graph nodes, and applies personalized PageRank to capture both topology structural and attribute similarities for community detection with the strategic game cost. GNN-based methods (e.g., GUCD [12], SDCN [13], DAEGC [19], SSGCAE [20], DNENC [21] and NMvC-GCN [22]) take the attribute vectors as the initial features of nodes, and realize the information fusion via feature propagation and aggregation operations of graph neural networks. Due to the strong nonlinear feature learning ability, these methods often can obtain good community detection performance, as long as combining the appropriate clustering model or training objective.

Generally, existing approaches have made a lot of efforts on fusing topology information and attribute information to uncover accurate and high quality communities. However, most of them still cannot fuse information adaptively. For example, SA-Cluster, DGTA, SCI, ANEM and AGNMF-AN all need to manually tune the corresponding hyperparameters to balance the contributions of different types of information. Furthermore, they pay little attention to extracting topics from communities. Although some methods, such as SCI and SA-Cluster, have made some efforts, they only extract the single topic from the community. In a word, existing methods still fail to comprehensively address these two challenges summarized in Section 1.

2.2. Graph self-supervised learning

Self-Supervised Learning (SSL), as a subset of the unsupervised learning, aims to learn representations via extracting supervision signals from the unlabeled data itself. This learning paradigm does not rely on any human-provided labels and has obtained great success in many research areas, including computer vision, signal processing and natural language processing [23–25]. Recently, inspired by these success, extensive researches focus on transferring SSL to graph-structured data, which promotes the popularity of graph SSL. Literatures [26–28] systematically categorize and review existing graph SSL methods, among of which two types of methods are dominant: contrast-based methods and generation-based methods. Contrast-based methods, such as DGI [29], GraphCL [30] and GCA [31], learn representations by making the representations of similar augmented objects (e.g., node, subgraph and graph) agree with each other, and those of dissimilar augmented objects disagree with each other. In such context, mutual information maximization is usually adopted to construct the objective function. Note that improper graph augmentation operations (e.g., drop or add edges/nodes) may result in the poor performance [32]. Without augmenting the graph, generation-based methods mainly leverage graph reconstruction loss as the supervision signal to learn representations. This type of methods are often based on the autoencoder-style framework composed of one or more encoders and decoders, such as GAE/VGAE [33], GraphMAE [34] and SIG-VAE [35]. Relatively speaking, this autoencoder-style framework is more flexible, because both the encoder and decoder can have various implementations, and their good combination often can boost the performance.

Essentially, community detection in attributed networks is unsupervised, so graph SSL methods can be naturally applied to it and some related method have been presented. Generally, most of them belong to two-step approaches, i.e., the representation learning and community detection are separated, such as GraphCL and VGAE. Some other methods are based on the end-to-end framework, which treats the representation learning (network embedding) and community detection as the unified optimization object, such as GUCD, SDCN and DAEGC. Inspired by these methods above, in this paper our proposed method SSAGCN also follows the graph SSL paradigm and adopts the autoencoder-style framework. However, compared to these existing methods, SSAGCN has at least two differences: (1) it incorporates the special information fusion strategy to boost the performance of the encoder. (2) it constructs a dual decoder for reconstructing topology structure and network attributes simultaneously, while other methods only reconstruct one of them.

3. Methodology

Throughout the rest of this paper, bold uppercase letters are used to denote matrices. For a given matrix $\mathbf{N}$, its ith row vector, entry (i, j), trace and transpose are respectively represented as $\mathbf{N}_i$, $\mathbf{N}_{ij}$, $tr(\mathbf{N})$ and $\mathbf{N}^T$.

3.1. Problem formulation

Here we specially pay attention to undirected attributed networks, and a given instance can be denoted as an undirected graph without weights $G = \{V, E, \mathbf{A}, \mathbf{X}\}$, where $V = \{v_1, v_2, \dots, v_n\}$ is the nodes set with size of n, and $E = \{e_{ij} | v_i \in V \wedge v_j \in V\}$ is the set of m links between nodes. The adjacency matrix $\mathbf{A} = [\mathbf{A}_{ij}]^{n\times n}$ is applied to represent the topology information of G. If $e_{ij} \in E$, $\mathbf{A}_{ij} = 1$, and $\mathbf{A}_{ij} = 0$ otherwise. The node-attribute matrix $\mathbf{X} = [\mathbf{X}_{iq}]^{n\times l}$ is employed to represent the attribute information of G. For the attributes set with size l $Y = \{y_1, y_2, \dots, y_l\}$, if v_i is associated with attribute y_q, $\mathbf{X}_{iq} = 1$, and $\mathbf{X}_{iq} = 0$ otherwise.

Given the aforementioned definitions, the problem of multiple topics community detection is to partition nodes in G into k disjoint clusters $C = \{c_1, c_2, \dots, c_k\}$ having d topics $Z = \{z_1, z_2, \dots, z_d\}$, so that (1) nodes within the same cluster have more dense links and common attributes than nodes in different clusters, and (2) each community c_i is associated with a topic distribution vector $\mathbf{S}_i = [\mathbf{S}_{i1}, \mathbf{S}_{i2}, \dots, \mathbf{S}_{id}]$, where $\mathbf{S}_{ij}$ represents the probability of topic z_j belonging to community c_i, and each topic can be represented through a l-dimensional attribute distribution vector.

3.2. SSAGCN method

3.2.1. Overview

To achieve the goal of fusing topology information and attribute information for multiple topics community detection, G is regarded as the topology graph $G_t = \{V, E_t, \mathbf{A}_t, \mathbf{X}\}$, where $E_t = E$ and $\mathbf{A}_t = \mathbf{A}$. Then we compute the cosine similarities between node attribute vectors in $\mathbf{X}$, and construct a K-nearest neighbor graph treated as the attribute similarity graph $G_a = \{V, E_a, \mathbf{A}_a, \mathbf{X}\}$, where E_a and $\mathbf{A}_a$ are respectively the edges set and the binary adjacent matrix of G_a. For G_t and G_a, their node representations $\mathbf{H}_t$ and $\mathbf{H}_a$ are respectively learned by two GCNs with shared weight parameters. Considering that G_t and G_a have homogeneity and heterogeneity information, $\mathbf{H}_t$ and $\mathbf{H}_a$ may include consensus and distinctive features. This means that they can play different roles in community detection, and it is important to learn their distinct contributions to multiple topics community detection. To this end, we introduce an attention mechanism that adaptively fuse $\mathbf{H}_t$ and $\mathbf{H}_a$ into the final node representation $\mathbf{H}$. Furthermore, in order to simultaneously drive $\mathbf{H}$ to preserve community structures and extract the topics of each community, we design a unified objective to train the combination of two GCNs and the attention network, called Adaptive GCN (AGCN) for simplification. This objective is composed of the modularity maximization and the losses of reconstructing topology structure and network attributes.

Following the aforementioned idea, we devise SSAGCN via the self-supervised learning framework with the autoencoder architecture (Fig. 1). As we can see, it mainly consists of three modules: AGCN encoder, Modularity maximization and Dual decoder, which are detailed in the following subsections.

3.2.2. AGCN encoder

As the encoder for learning node representation, AGCN comprises two GCNs with shared weight parameters. One GCN takes the topology graph G_t as the input, and learns the node representation from G_t via:

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

where $\mathbf{W}^{(L)}$ and $\mathbf{H}_t^{(L)}$ are respectively the trainable weight matrix and node representation matrix of the Lth layer in GCN. $\widetilde{\mathbf{A}}_t = \mathbf{A}_t + \mathbf{I}$ is the adjacent matrix of G_t with self-loop, and $\widetilde{\mathbf{D}}_t$ is its diagonal degree matrix. $\mathrm{Relu}(x) = \max(0, x)$ is the nonlinear activation function. We set $\mathbf{H}_t^{(0)} = \mathbf{X}$, and the node representation matrix of the final layer is denoted as $\mathbf{H}_t$. Following the same way, the other GCN sharing $\mathbf{W}^{(L)}$

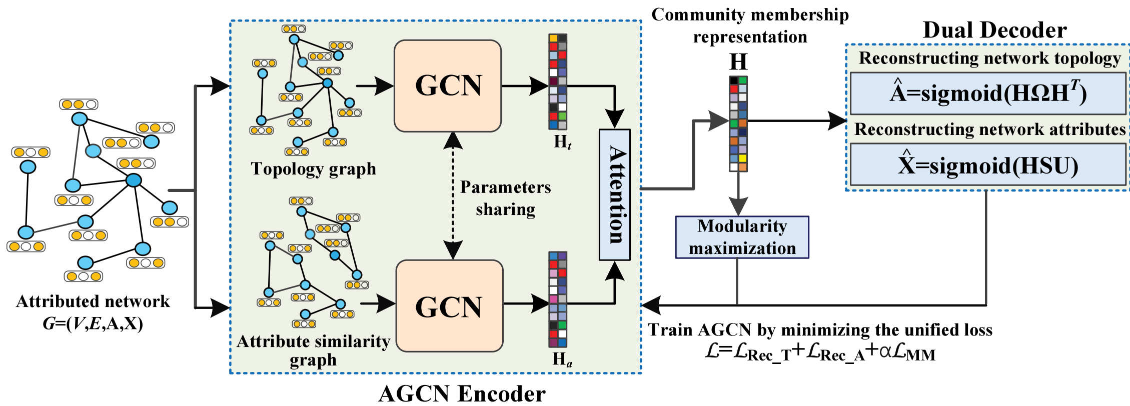


Fig. 1. The overall framework of SSAGCN. $\mathbf{H}$, $\mathbf{S}$ and $\mathbf{U}$ respectively denote the community membership matrix, community-topic matrix and topic-attribute matrix, and $\mathcal{L}_{\text{Rec\_T}}$, $\mathcal{L}_{\text{Rec\_A}}$ and $\mathcal{L}_{\text{MM}}$ respectively denote the losses of reconstructing topology structure, reconstructing network attributes and modularity maximization. Note that the dimensions of $\mathbf{H}_t$, $\mathbf{H}_a$ and $\mathbf{H}$ are uniformly set to $n \times k$.

takes the attribute similarity graph G_a as the input, and learns the node representation from G_a via:

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

where $\mathbf{H}_a^{(0)} = \mathbf{X}$ and the last layer output representation is denoted as $\mathbf{H}_a$. To adaptively fuse $\mathbf{H}_t$ and $\mathbf{H}_a$ for the downstream task of community detection, a special attention mechanism is designed to learn their respective importance: $(\delta_t, \delta_a) = \text{Attention}(\mathbf{H}_t, \mathbf{H}_a)$, where δ_t, $\delta_a \in \mathbb{R}^{n\times 1}$. For $\mathbf{H}_t$, we firstly apply a nonlinear activation function (e.g., LeakyReLU) to transform it, and then introduce a shared attention matrix $\mathbf{\Theta} \in \mathbb{R}^{1\times k'}$ to learn its attention weight $\omega_t \in \mathbb{R}^{n\times 1}$:

$$\omega_t = \text{LeakyRelu}(\mathbf{H}_t\mathbf{W}_t^T + \mathbf{B}_t)\mathbf{\Theta}^T, \tag{3}$$

where $\mathbf{W}_t \in \mathbb{R}^{k'\times k}$ and $\mathbf{B}_t \in \mathbb{R}^{n\times k'}$ are the weight matrix and bias matrix, respectively. In a similar way, we can calculate the attention weight $\omega_a \in \mathbb{R}^{n\times 1}$ of $\mathbf{H}_a$. By using the softmax function we normalize ω_t to obtain δ_t:

$$\delta_t = \text{softmax}(\omega_t) = \frac{\exp(\omega_t)}{\exp(\omega_a) + \exp(\omega_t)}. \tag{4}$$

Similarly, $\delta_a = \text{softmax}(\omega_a)$. Let $\boldsymbol{\delta}_t = \text{diag}(\delta_t) \in \mathbb{R}^{n\times n}$ and $\boldsymbol{\delta}_a = \text{diag}(\delta_a) \in \mathbb{R}^{n\times n}$, then the final node representation $\mathbf{H}$ is obtained via:

$$\mathbf{H} = \boldsymbol{\delta}_t\mathbf{H}_t + \boldsymbol{\delta}_a\mathbf{H}_a. \tag{5}$$

3.2.3. Modularity maximization module

To enhance the representation of community membership, we introduce the modularity maximization metric [36], a widely used measure for modeling community structure, to refine $\mathbf{H}$. Specifically, by treating $\mathbf{H}$ as the matrix of community memberships, we approximate the modularity Q of G as follows:

$$Q = \frac{1}{4m}\sum_{i,j}^{n}(\mathbf{A}_{ij} - \frac{d_i d_j}{2m})\mathbf{H}_i\mathbf{H}_j^T, \tag{6}$$

where d_i is the degree of v_i. Let $\mathbf{M} = [\mathbf{M}_{ij}]^{n\times n}$ whose element $\mathbf{M}_{ij} = \mathbf{A}_{ij} - \frac{d_i d_j}{2m}$, and omitting $\frac{1}{4m}$ which does not affect the maximum of Q, then Q can be simplified as its trace form:

$$Q = tr(\mathbf{H}^T\mathbf{M}\mathbf{H}) \quad s.t., \sum_{j=1}^{k}\mathbf{H}_{ij} = 1. \tag{7}$$

To train AGCN, we select $-Q$ as the loss function $\mathcal{L}_{\text{MM}}$, which can guide AGCN to learn more discriminative community memberships representation $\mathbf{H}$. In our experiments, we will specially analyze the effect of modularity maximization module.

3.2.4. Dual decoder

As shown in Fig. 1, the dual decoder comprises two decoders respectively utilized to reconstruct topology structure and attributes. Inspired by the community detection model based on nonnegative matrix tri-factorization (NMTF) [37], the decoder for reconstructing the topology structure is designed as $\mathbf{A} \approx \mathbf{H}\mathbf{\Omega}\mathbf{H}^T$, where $\mathbf{\Omega}$ is the nonnegative community interaction matrix, and $\mathbf{H}$ is also nonnegative, which can be inferred from Eq. (5). By further combining NMTF with the sigmoid activation function, we apply it to reconstruct the topology structure $\hat{\mathbf{A}} = [\hat{\mathbf{A}}_{ij}]^{n\times n}$ as follows:

$$\hat{\mathbf{A}} = \text{sigmoid}(\mathbf{H}\mathbf{\Omega}\mathbf{H}^T) \quad s.t., \mathbf{\Omega} \geq 0. \tag{8}$$

To make $\mathbf{\Omega}$ become the learnable parameter, we can treat this NMTF model as a decoder neural network. Letting $\hat{\mathbf{A}}$ approximate to the original topology structure matrix $\mathbf{A}$, we measure the loss of reconstructing topology structure via the binary cross entropy function:

$$\mathcal{L}_{\text{Rec\_T}} = -\frac{1}{n^2}\sum_{i,j}^{n}(\mathbf{A}_{ij}\log\hat{\mathbf{A}}_{ij} + (1-\mathbf{A}_{ij})\log(1-\hat{\mathbf{A}}_{ij})). \tag{9}$$

For the decoder used to reconstruct network attributes, we borrow the idea of LDA topic model [38] to extract topics from community. Specifically, every node is denoted as the probability distribution over k communities, every community is denoted as the probability distribution over d topics and every topic is denoted as the probability distribution over l attributes. Then we can use the generative model shown in Fig. 2 to drive the process of the attribute reconstruction. More precisely, assuming that every node-attribute pair (v_i, y_q) is independently generated, we can denote the probability of v_i having attribute y_q as:

$$\hat{\mathbf{X}}_{iq} = P(y_q|v_i) = \sum_{z_p\in Z}\sum_{c_j\in C}P(y_q|z_p)P(z_p|c_j)P(c_j|v_i), \tag{10}$$

where $P(c_j|v_i)$ represents the probability of node v_i belonging to community c_j, $P(z_p|c_j)$ indicates the probability of community c_j being associated with topic z_p, and $P(y_q|z_p)$ denotes the probability of topic z_p having attribute y_q. Letting $P(c_j|v_i) = \mathbf{H}_{ij}$ and $\hat{\mathbf{X}} = [\hat{\mathbf{X}}_{iq}]^{n\times l}$, $\mathbf{S} = [\mathbf{S}_{jp}]^{k\times d}$ and $\mathbf{U} = [\mathbf{U}_{pq}]^{d\times l}$ respectively denote the community-topic and topic-attribute matrices, we can rewrite Eq. (10) as the matrix form:

$$\hat{\mathbf{X}} = \text{sigmoid}(\mathbf{H}\mathbf{S}\mathbf{U}) \quad s.t., \sum_{p=1}^{d}\mathbf{S}_{jp} = 1, \sum_{q=1}^{l}\mathbf{U}_{pq} = 1. \tag{11}$$

We take this model shown in Eq. (11) as two layers of the decoder neural network used to reconstruct network attributes. Through this

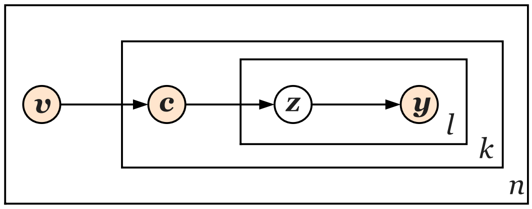


Fig. 2. The generative model for extracting topics. The observed variables v, c and y respectively denote the node, community and attribute, and the latent variable z denotes the topic.

way, **S** and **U** are turned into the trainable parameters. From the probability perspective, the generation of attribute matrix **X** can be denoted as:

$$P(\mathbf{X}|V) = \prod_{i=1}^{n}\prod_{q=1}^{l}(\hat{\mathbf{X}}_{iq})^{\mathbf{X}_{iq}}. \tag{12}$$

Maximizing the likelihood of $P(\mathbf{X}|V)$ can be the objective of network attributes generative model, and hence we treat the negative log-likelihood of $P(\mathbf{X}|V)$ as the loss of network attributes reconstruction:

$$\mathcal{L}_{\mathrm{Rec\_A}} = -\sum_{i=1}^{n}\sum_{q=1}^{l}\mathbf{X}_{iq}\log\hat{\mathbf{X}}_{iq}. \tag{13}$$

3.2.5. The unified objective

To train all learnable parameters in SSAGCN, we integrate $\mathcal{L}_{\mathrm{Rec\_T}}$, $\mathcal{L}_{\mathrm{Rec\_A}}$ and $\mathcal{L}_{\mathrm{MM}}$ into the unified objective function:

$$\mathcal{L} = \mathcal{L}_{\mathrm{Rec\_T}} + \mathcal{L}_{\mathrm{Rec\_A}} + \alpha\mathcal{L}_{\mathrm{MM}}, \tag{14}$$

where hyperparameter α is used to balance the influence of the modularity maximization module. It is worth noting that we do not specify weights for $\mathcal{L}_{\mathrm{Rec\_T}}$ and $\mathcal{L}_{\mathrm{Rec\_A}}$. The reason is that we have integrated the adaptive information fusion mechanism (refer to Section 3.2.2) into the AGCN encoder. This mechanism autonomously learns the respective weights of topology and attribute information. Since both $\mathcal{L}_{\mathrm{Rec\_T}}$ and $\mathcal{L}_{\mathrm{Rec\_A}}$ are designed to reconstruct topology structure and network attributes through the generative models, this objective function offers a self-supervised way for training the AGCN encoder. Moreover, $\mathcal{L}_{\mathrm{MM}}$ drives the AGCN encoder towards community detection, making the overall SSAGCN framework an end-to-end solution for detecting communities with multiple topics.

3.2.6. Multiple topics community detection

We select $\mathcal{L}$ to train AGCN and use backward propagation algorithm to update its related weight parameters. When the training process converges or reaches the maximum iterations, we can determine the community index r of v_i by:

$$r = \underset{j=1\ldots k}{\operatorname{argmax}}\,\mathbf{H}_{ij}. \tag{15}$$

For community c_r, we can extract its affiliated multiple topics from **S** by presetting a threshold value φ. Specifically, if $\mathbf{S}_{rp} > \varphi$, topic t_p will be associated with c_r. Besides, the corresponding key attributes of t_p can be extracted by sorting the elements of $\mathbf{U}_p$ in descending order. To better demonstrate the working process of multiple topics community detection, by following the flowchart idea presented in [39] we depict the overall process of SSAGCN in Fig. 3. We can observe that the computational cost of SSAGCN is mainly concentrated in three parts: AGCN encoder, modularity maximization module and dual decoder. Their time complexities are $O(ml + nl^2 + nkk')$, $O(n^2k + nk^2)$ and $O(nk^2 + n^2k + n + nkd + ndl)$, respectively. In total, the approximate time complexity of SSAGCN is $O(MaxIter(ml + nl^2 + nkk' + n^2k + nk^2 + nkd + ndl))$, where $MaxIter$ denotes the maximum iterations during the training.

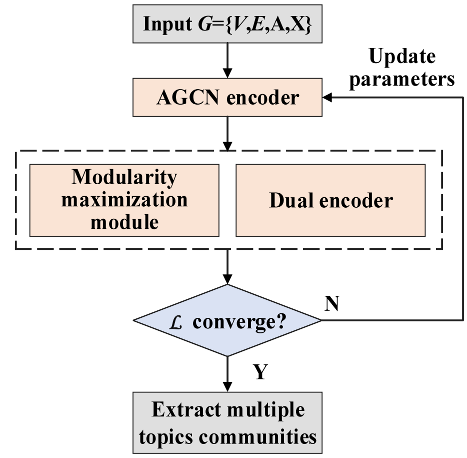


Fig. 3. The flowchart of multiple topics community detection using SSAGCN.

Table 1
Statistics of datasets.

Datasets	Type	n	m	l	k
Texas	Hyperlink	187	747	1703	5
Washington	Hyperlink	230	962	1703	5
Wisconsin	Hyperlink	265	1183	1703	5
Cora	Citation	2708	5429	1433	7
Citeseer	Citation	3312	4732	3703	6
UAI2010	Citation	3363	45,006	4972	19
Pubmed	Citation	19,729	44,338	500	3
Blogcatalog	Social	5196	171,743	8189	6
Flickr	Social	7575	239,738	12,047	9

4. Experiments

In this section, we validate whether SSAGCN is enough effective on various types of attribute networks, and the running platform is a PC with 64-bit Windows 10 system, 3.5 GHz Intel Core i9-11900K CPU and 64 GB RAM. The program language is Python 3.7 and GNNs in all methods are developed using Deep Graph Library (DGL).[1]

4.1. Experiment setup

(1) Benchmark datasets. We select 9 attributed networks that are widely used and have ground-truth community labels as benchmark datasets. These networks include three hyperlink networks from WebKB dateset[2]: Texas, Washington, Wisconsin, four citation networks: Cora [40], Citeseer [40], UAI2010 [13] and Pubmed [40], and two online social networks: Blogcatalog [41] and Flickr [41]. Nodes and links in hyperlink networks respectively represent webpages and links between webpages, nodes and links in citation networks respectively represent papers and citation relationships between papers, and nodes and links in social networks respectively represent users and interactive relationships between users. Each node in these networks has a binary value (0 or 1) attribute vector. The detail statistics of these networks are shown in Table 1.

(2) Baseline methods. As discussed in Sections 1 and 2, compared to those utilizing only topology information or attribute information, community detection methods that are capable of integrating these two

[1] https://www.dgl.ai.

[2] https://www.cs.umd.edu/~sen/lbc-proj/LBC.html.

types of information often yield better performance. In view of this, we only select approaches simultaneously utilizing topology information and attribute information as baselines, including four types of methods: GNN-based methods (GUCD [13], SDCN [12] and DAEGC [19]), probabilistic generative model-based method (AdaMRF [7]), game theory-based methods (DGTA [9], NMF-based SCI [3] and ANEM [5]), and SSL-based methods (GraphCL [30] and VGAE [33]). These methods are all representatives and their features have been outlined in Section 2. Note that GraphCL and VGAE are not end-to-end methods, and thus we additionally use the k-means clustering model to detect communities based on their final output representations. Besides, to validate the effectiveness of AGCN in SSAGCN, we replace it with the original GCN [42] that directly takes the attribute vectors as the initial features, and call this version SSGCN, which is also selected as one of baselines.

(3) Evaluation metrics. To provide the comprehensive evaluation to SSAGCN, we choose four widely used evaluation metrics for community detection in attributed networks, including Normalized Mutual Information (NMI) [43], Adjusted Rand Index (ARI) [43], Modularity Q in Eq. (6) and Average Entropy (AE). Both NMI and ARI metrics are used to evaluate the accuracy of community detection. Modularity Q is applied to test the cohesiveness of communities. AE is utilized to measure the degree of attribute confusion between nodes in the same community, and its definition is as follows:

$$\mathrm{AE}(\{c_i\}_{i=1}^{k}) = \sum_{i=1}^{k}\sum_{j=1}^{l}\frac{|c_i|}{|V|}entropy(c_i, y_j), \tag{16}$$

where $entropy(c_i, y_j) = -p_{ij}\log p_{ij}$, and p_{ij} is the percentage of nodes with attribute y_j in community c_i. According to the requirement of community detection in attributed networks defined in Section 3, nodes in the same community should not only link to each other more densely, but also share more common attributes. Thus the larger the Q is and the smaller the AE is, the better the performance of community detection will be. Besides, if the scores of NMI and ARI are larger, the performance will also be better.

(4) Parameter settings. For our proposed method SSAGCN, the layer configuration of its AGCN encoder is set to l-128-k across all datasets, since we find that more layers will lead to the over-smoothing problem as reported in many literatures. To facilitate the analysis, we set $d = k$. Namely, the topics number is the same to the number of the known community labels. However, it should be pointed out that d and k can be different, which will be specially validated in Section 4.3 by conducting a case study of SIGIR co-authorship network. The balance parameter α in Eq. (14) is set to 1, and K, determining how many nearest neighbors should be selected to construct the attribute similarity graph, is set to: 10 on Texas, Washington, Wisconsin, Cora, Citeseer and UAI2010, 30 on Pubmed and 15 on Blogcatalog and Flickr. We provide the deep analysis to the settings of α and K in Section 4.2.(5). The Adam optimizer with learning rate of 0.001 is utilized to train AGCN. For SSGCN, we adopt the same settings as SSAGCN. For other baseline methods, we retain the configurations recommended in their respective papers. In order to guarantee the fair comparison, every method is run 10 times and its average performance is reported here.

4.2. Results and analysis

(1) Comparison with baselines. On each dataset, we conduct a comprehensive comparison of SSAGCN with baseline methods, and the results are summarized in Table 2. It is evident that SSAGCN outperforms all baselines on most datasets, except for the Citeseer network, where it still achieves the runner-up performance in NMI, ARI and AE. In particular, on Blogcatalog and Flickr SSAGCN exhibits remarkable advantages. On Blogcatalog, compared to the runner-up results, SSAGCN respectively achieves the improvements of 26.4%, 12.1%, 16.6% and 7.58% in NMI, ACC, Q and AE. Similarly, on Flickr the corresponding improvements are respectively 12.3%, 19.1%, 6.18% and 9.95%. Besides, we also find that, compared to other baselines, SSGCN without using AGCN encoder does not show any advantages. These results illustrate that SSAGCN is effective and superior, which may attribute to its two features: adaptive information fusion and multiple topics community detection, and that other baselines do not have these two features simultaneously. In the following subsections, we will further analyze the functions of these two features.

(2) Adaptive information fusion analysis. For fusing topology information and attribute information that are respectively encoded as $\mathbf{H}_t$ and $\mathbf{H}_a$, we design the attention mechanism to adaptively learn their respective weights in SSAGCN. To assess its effectiveness, we first conduct an analysis of the attention distribution across all datasets, with the results presented in Fig. 4. We can observe that: topology information is important than attribute information on Texas, Washington, Wisconsin, Cora, Citeseer and Blogcatalog, but it is opposite on UAI2010, Pubmed and Flickr. In Fig. 5, we further illustrate the changing trends of the average attention values during the training process. It is clear that the attention values for topology information and attribute information are quite similar in the initial stage, but they are different as the training epochs progress. For instance, on Cora the attention value of topology information increases gradually, while that of attribute information is always decreasing. Specially, on Blogcatalog the attention value of topology information decreases at the beginning, but it begins to increase after 25 epochs and finally surpass that of attribute information, which is consistent with Fig. 4(h). Similar phenomenon can also be found on Washington in Fig. 5(b) and UAI2010 in Fig. 5(f). These findings sufficiently demonstrate the function of the attention mechanism used in SSAGCN, that can automatically learn the respective importance of topology information and attribute information during the training.

In the other aspect, the adaptive information fusion is able to improve the robustness against noises. To validate this, we select randomly ρ% node pairs without links across different communities and add the links to them. With the increasing of ρ, community structure will be more and more unclear, resulting in the task of community detection becomes increasingly difficult. Here, we take Cora as the example, select all baselines as the competitors and evaluate the performance by changing ρ from 10 to 50 with a step length of 10. The results are shown in Fig. 6, from which we can find that: all methods tend to deteriorate when ρ increases, but SSAGCN still performs the best at any value of ρ. More importantly, the performance of SSAGCN is surprisingly stable and does not decrease sharply like other approaches: when ρ changes from 0 to 50, SSAGCN respectively decreases by 13.1%, 6.99% and 10.1% in terms of NMI, ARI and Q, and the AE score increases by 13.6%, but for baselines their NMI, ARI and Q scores at least respectively decrease by 34.7%, 44.1% and 26.9%, and the AE score at least increases by 23.4%. These results illustrate that SSAGCN has better robustness against noises than baselines, which also means that the mechanism of adaptive information fusion enables SSAGCN to fully exploit useful information from both topology structure and attributes.

(3) Multiple topics community detection. To intuitively illustrate whether SSAGCN can detect communities with multiple topics, the community-topic matrices (i.e., $\mathbf{S}$) of all datasets are visualized in Fig. 7. We can observe that community and topic has a clear many to many relationship. For instance, on Cora community c_1 exhibits strong correlations with topics t_5 and t_6, and on Flickr community c_5 are strongly associated with topics t_2 and t_3. In the real world, it is quite common that papers (e.g., nodes in Cora, Citeseer and Pubmed) in the same community (cluster) have multiple identical research topics. Likewise, online social networks users (e.g., nodes in Blogcatalog and Flickr) often share multiple common topics of interest in the same user group. The visualization results on these networks well exhibit these phenomenons. In Section 4.3, we will further show the good ability of

Table 2
Performance comparison results with baseline methods. Bold and underlined respectively indicate the best and the runner-up.

Datasets	Metrics	GUCD	SDCN	DAEGC	AdaMRF	DGTA	SCI	ANEM	GraphCL	VGAE	SSGCN	SSAGCN
Texas	NMI	0.275	0.202	0.271	0.255	0.207	0.261	0.236	0.201	0.261	0.213	**0.281**
	ARI	0.332	0.326	0.322	0.253	0.225	0.293	0.261	0.287	0.221	0.223	**0.349**
	Q	0.381	0.366	0.331	0.371	0.369	0.385	0.363	0.361	0.382	0.381	**0.416**
	AE	12.273	12.347	11.358	13.276	13.729	13.761	12.756	13.763	13.711	12.367	**9.203**
Washington	NMI	0.315	0.383	0.393	0.371	0.366	0.379	0.353	0.302	0.377	0.362	**0.421**
	ARI	0.331	0.372	0.407	0.349	0.378	0.352	0.337	0.283	0.403	0.401	**0.433**
	Q	0.256	0.291	0.289	0.288	0.246	0.271	0.203	0.273	0.265	0.269	**0.306**
	AE	12.369	13.002	10.789	12.116	12.735	13.057	12.473	14.201	13.257	11.537	**9.327**
Wisconsin	NMI	0.305	0.327	0.351	0.346	0.331	0.343	0.308	0.281	0.319	0.327	**0.376**
	ARI	0.226	0.254	0.282	0.273	0.277	0.261	0.213	0.206	0.243	0.266	**0.293**
	Q	0.352	0.331	0.336	0.369	0.346	0.339	0.352	0.335	0.351	0.357	**0.377**
	AE	11.221	12.017	10.782	10.993	12.005	11.633	12.846	12.331	13.019	10.943	**9.133**
Cora	NMI	0.323	0.491	0.528	0.362	0.451	0.382	0.345	0.275	0.478	0.513	**0.565**
	ARI	0.319	0.413	0.496	0.304	0.431	0.343	0.304	0.209	0.435	0.473	**0.515**
	Q	0.557	0.562	0.649	0.522	0.623	0.701	0.499	0.487	0.707	0.698	**0.724**
	AE	14.725	12.323	10.679	14.003	12.881	13.727	14.166	15.863	12.602	10.942	**9.017**
Citeseer	NMI	0.274	0.387	**0.397**	0.288	0.217	0.383	0.281	0.188	0.218	0.382	0.391
	ARI	0.232	**0.402**	0.378	0.273	0.119	0.312	0.262	0.173	0.121	0.367	0.379
	Q	0.555	0.572	0.549	0.556	0.512	**0.579**	0.554	0.431	0.503	0.553	0.561
	AE	11.251	10.469	**10.223**	11.791	12.328	10.331	11.883	13.012	12.319	10.919	10.312
UAI2010	NMI	0.321	0.309	0.303	0.313	0.286	0.312	0.334	0.296	0.341	0.331	**0.343**
	ARI	0.312	0.306	0.301	0.305	0.283	0.308	0.303	0.302	0.313	0.298	**0.315**
	Q	0.473	0.433	0.427	0.431	0.369	0.435	0.349	0.387	0.446	0.451	**0.475**
	AE	11.925	11.889	11.695	11.551	12.752	11.78	11.849	12.661	11.232	12.003	**11.226**
Pubmed	NMI	0.269	0.253	0.243	0.211	0.231	0.261	0.234	0.219	0.266	0.241	**0.327**
	ARI	0.211	0.202	0.217	0.209	0.205	0.201	0.204	0.211	0.213	0.193	**0.288**
	Q	0.412	0.488	0.505	0.517	0.522	0.508	0.402	0.518	0.527	0.483	**0.607**
	AE	9.231	9.656	9.878	10.853	10.171	9.337	10.161	10.662	9.212	9.591	**8.414**
Blogcatalog	NMI	0.302	0.311	0.277	0.289	0.256	0.302	0.308	0.188	0.253	0.282	**0.393**
	ARI	0.231	0.216	0.202	0.223	0.221	0.213	0.212	0.143	0.166	0.203	**0.259**
	Q	0.142	0.171	0.238	0.232	0.231	0.241	0.235	0.138	0.191	0.221	**0.245**
	AE	12.541	11.947	13.251	13.072	14.219	12.338	12.012	14.975	14.335	12.366	**11.041**
Flickr	NMI	0.271	0.359	0.308	0.253	0.239	0.357	0.313	0.156	0.178	0.341	**0.403**
	ARI	0.209	0.223	0.215	0.206	0.227	0.231	0.211	0.121	0.109	0.212	**0.275**
	Q	0.175	0.157	0.177	0.171	0.173	0.178	0.123	0.127	0.172	0.163	**0.189**
	AE	14.223	13.594	13.981	14.972	15.266	13.761	13.926	15.743	15.002	13.997	**12.203**

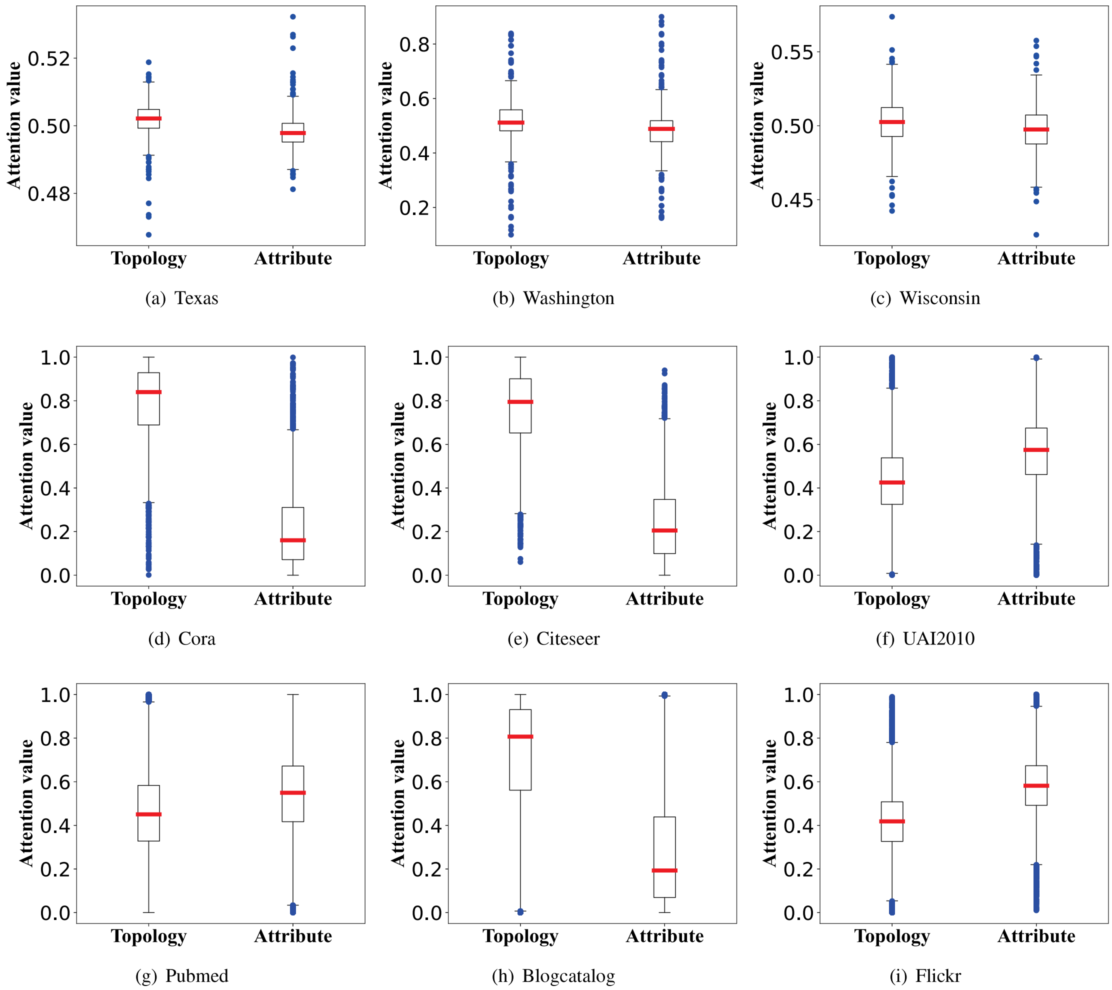
SSAGCN for multiple topics community detection through a practical case study on SIGIR co-authorship network.

(4) Ablation study. As mentioned above, SSAGCN consists of two decoders: reconstructing topology structure and reconstructing network attributes. Then, can we utilize only one decoder? To answer this question, on all datasets we specially compare SSAGCN with its two variants: SSAGCN with reconstructing topology structure decoder only (SSAGCN-Rec_T) and SSAGCN with reconstructing network attributes decoder only (SSAGCN-Rec_A). The results are presented in Table 3. Considering Q and AE have the same changing trends, here we only give the results of the NMI and ARI. It is clear that: (1) SSAGCN consistently performs better than two variants, indicating the effectiveness of using the dual decoder. However, its advantages are not obvious. This means that compared to the dual decoder, the modularity maximization module plays a leading role in terms of boosting the performance. In the next subsection, we will specially investigate the importance of the modularity maximization module. (2) On UAI2010, Pubmed and Flickr SSAGCN-Rec_A performs slightly worse than SSAGCN-Rec_T, but performs slightly better than SSAGCN-Rec_T on other datasets. The reason for this may be related to the different importance of topology information and attribute information on different datasets. In summary, if we use only one decoder, the performance just decreases slightly. Different decoders have their own advantages, but we should note that the reconstructing attributes decoder can help us to uncover the community distribution on topics, which has been demonstrated in Section 4.2.(3).

(5) Parameter sensitivity analysis. To investigate the impact of the top K neighbors in attribute similarity graph on every dataset, we

Table 3
Performance comparison results with variants having only one decoder. Bold and underlined respectively indicate the best and the runner-up.

Datasets	Metrics	SSAGCN-Rec_T	SSAGCN-Rec_A	SSAGCN
Texas	NMI	0.277	0.275	**0.281**
	ARI	0.341	0.340	**0.349**
Washington	NMI	0.413	0.409	**0.421**
	ARI	0.427	0.423	**0.433**
Wisconsin	NMI	0.373	0.371	**0.376**
	ARI	0.288	0.285	**0.293**
Cora	NMI	0.557	0.552	**0.565**
	ARI	0.507	0.505	**0.515**
Citeseer	NMI	0.388	0.386	**0.391**
	ARI	0.373	0.370	**0.379**
UAI2010	NMI	0.337	0.341	**0.343**
	ARI	0.309	0.311	**0.315**
Pubmed	NMI	0.321	0.325	**0.327**
	ARI	0.283	0.286	**0.288**
Blogcatalog	NMI	0.389	0.385	**0.393**
	ARI	0.254	0.252	**0.259**
Flickr	NMI	0.397	0.399	**0.403**
	ARI	0.268	0.271	**0.275**



(a) Texas (b) Washington (c) Wisconsin (d) Cora (e) Citeseer (f) UAI2010 (g) Pubmed (h) Blogcatalog (i) Flickr

Fig. 4. Analysis of attention distribution. The red horizontal lines denote the median values, and the blue dots above/under the box denote the excessively large/small values.

vary K from 2 to 50 and test the performance. The NMI and ARI results are presented in Fig. 8. For all datasets, we can find that the NMI and ARI scores increase first and then start to decrease when K reaches a certain value. It is because larger K may introduce more noisy links, which can reduce the quality of attribute similarity graph. According to these empirical results, for Texas, Washington, Wisconsin, Cora, Citeseer, UAI2010, Pubmed, Blogcatalog and Flickr, we respectively set K to 10, 10, 10, 10, 10, 10, 30, 15 and 15.

As for the trade off parameter α in the unified loss $\mathcal{L}$, we vary it in the range of $\{0, 1e^{-5}, 1e^{-4}, 1e^{-3}, 1e^{-2}, 1e^{-1}, 1, 1e^{1}, 1e^{2}, 1e^{3}, 1e^{4}, 1e^{5}\}$ and test SSAGCN in terms of NMI and ARI on every dataset. The results are presented in Fig. 9 and it can be seen that: on every dataset SSAGCN performs very poorly when $\alpha = 0$, but all consistently achieve the stable and best results when α is over 0. This means that α is not sensitive when it is over 0. Furthermore, it also indicates that the modularity maximization module is very important for SSAGCN. Actually, the modularity maximization module is the key to make SSAGCN become an end-to-end community detection method. To further verify this, we respectively set $\alpha = 0$ and $\alpha = 1$, and then map the learned community membership representation $\mathbf{H}$ into two-dimensional space through t-SNE [44]. As the representatives, Figs. 10 and 11 respectively presents the visualization results on Cora and Blogcatalog. We can observe more clearer boundaries among clusters (communities) when $\alpha = 1$. These results above fully demonstrate the powerful guiding effect of modularity maximization on community detection.

(6) Efficiency analysis. We select all baselines as the competitors, and conduct the running time comparison on three large-scale datasets: Pubmed, Blogcatalog and Flickr. The training epoches of all methods are uniformly set to 100, and the results are given in Table 4. It can be seen that: (1) SSAGCN takes more time on networks with more nodes, which is consistent with the result of the time complexity analysis in Section 3.2.6: the number of nodes dominates the time cost. (2) Compared to the best method, SSAGCN does not add too much time, especially on Blogcatalog and Flickr. Considering the advantages in terms of NMI, ARI, Q and AE analyzed above, SSAGCN is still very competitive in general.

4.3. A case study on SIGIR co-authorship network

As previously highlighted, SSAGCN is able to discover multiple topics communities in attributed networks and identify key attributes related to the corresponding topics, which means that it can uncover semantic communities. To provide further demonstration of this capability, we specially conduct a case study on SIGIR co-authorship network, which is constructed by extracting the co-authorships from SIGIR articles published between 2013 and 2022. This network comprises 5830 nodes, 48 252 links and 3901 attributes. We first set the optimal

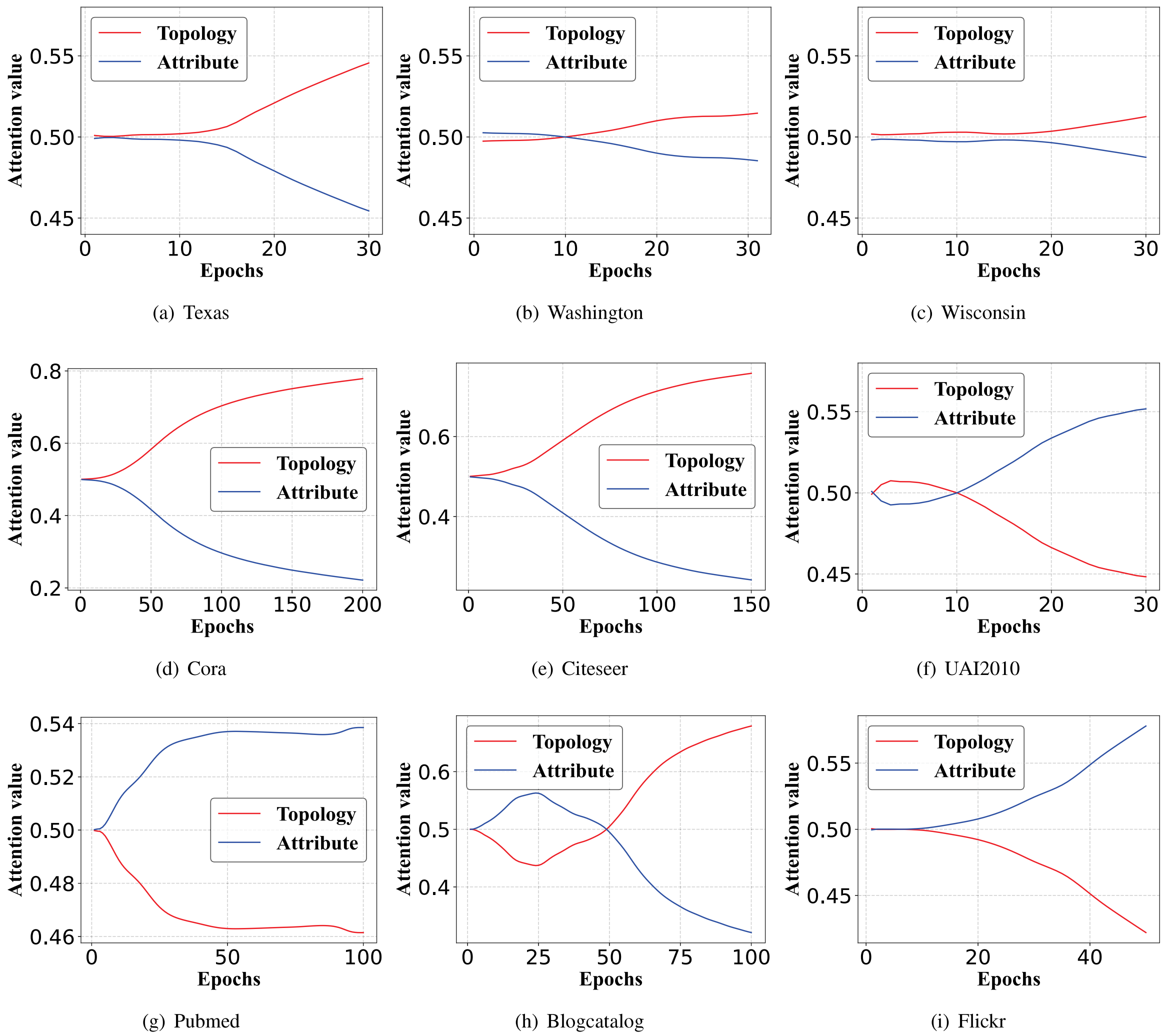


Fig. 5. The changing trends of the average attention values during the training.

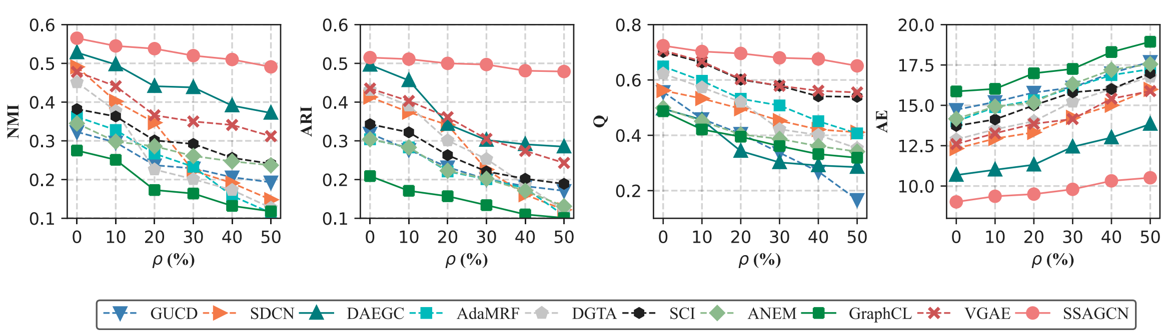
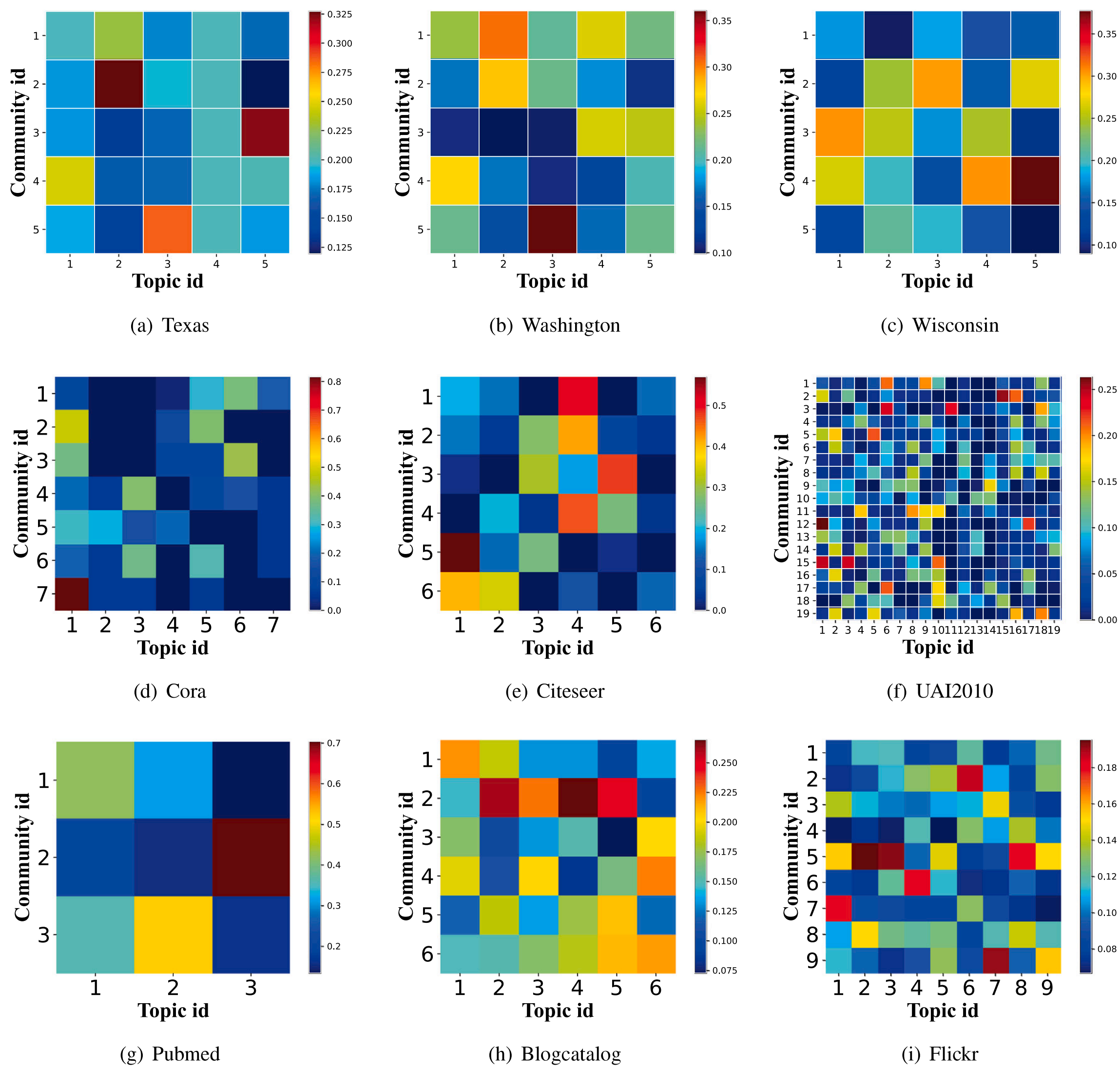


Fig. 6. Performance comparison on Cora w.r.t. the rate of noise links ρ%.

Table 4
Running time (in Seconds) comparison results with baselines on large-scale datasets. Bold and underlined respectively indicate the best and the runner-up.

Datasets	GUCD	SDCN	DAEGC	AdaMRF	DGTA	SCI	ANEM	GraphCL	VGAE	SSGCN	SSAGCN
Pubmed	415.43	**49.9**	2155.83	1863.21	1537.23	933	639.32	567.32	575.26	173.65	213.53
Blogcatalog	186.59	71.07	182.51	163.79	155.97	416	170.9	197.52	**65.82**	88.61	107.89
Flickr	414.25	152.26	2354.53	2038.33	2137.54	960	249.63	507.19	**136.23**	180.35	199.76



(a) Texas (b) Washington (c) Wisconsin
(d) Cora (e) Citeseer (f) UAI2010
(g) Pubmed (h) Blogcatalog (i) Flickr

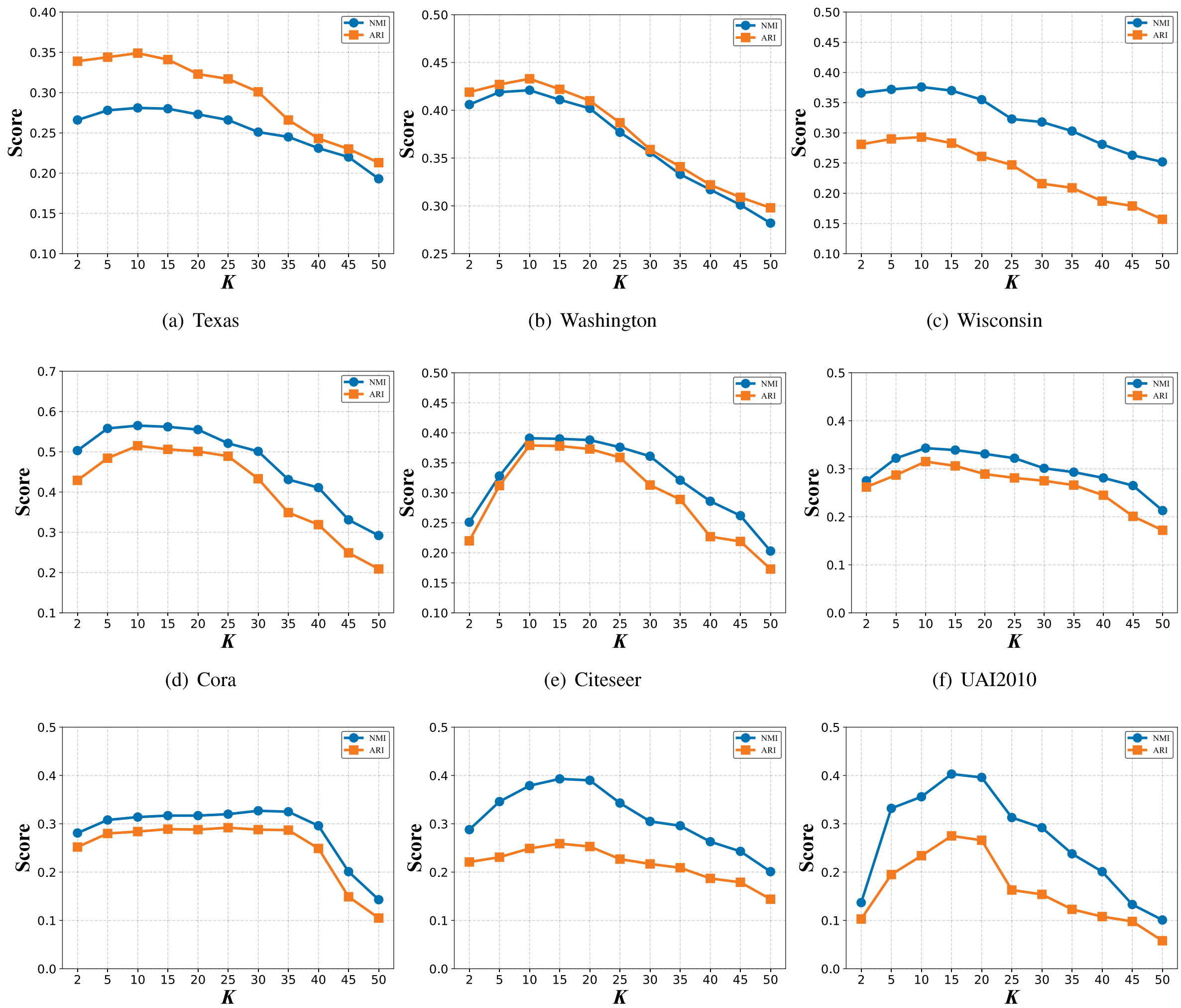
Fig. 7. The visualization of community-topic matrix. Different colors represent different strengths of the community-topic relationship.

number of communities to 451, which is determined via applying the classical community detection algorithm Louvain [45]. Subsequently, we set the number of topics t to 23 based on the perplexity measure of the LDA model [38]. It is well known that a distinguished scholar often leads a big research team having multiple research directions, which means that his/her community will be associated with multiple topics. Based on this, we take two famous IR scholars "Maarten de Rijke" and "W. Bruce Croft" as examples, and conduct the corresponding analysis. Their ego-centric community networks and community-topic distributions are respectively plotted in Figs. 12 and 13.

As we can see, both communities have dense internal links. More importantly, they have multiple prominent topics. For example, the community of Maarten de Rijke is obviously associated with topic t_7 and t_{20}, and those of W. Bruce Croft are topic t_{19} and t_{22}. Besides, we notice that W. Bruce Croft's community is also associated with t_7 strongly, which confirms the many-to-many relationship between community and topic. For these two communities, we further respectively list their core members (Table 5), and the key attributes of their strongly correlated topics (Table 6). It is clear that both Maarten de Rijke and W. Bruce Croft are the absolute cores of their respective communities. Besides, from the key attributes we can infer that t_7 and t_{20} are respectively about Search and Ranking, and t_{19} and t_{22} are respectively Recommendation and Question-Answering. These topics and associated key attributes can well display the research focuses of these communities.

5. Conclusions

In this paper, we propose SSAGCN, a method based on GNN and self-supervised learning, aiming to tackle two key challenges for community detection in attributed networks: information fusion and discovering communities with multiple topics. SSAGCN has two prominent features: adaptive fusion of topology information and attribute information, and learning communities with multiple topics in an end-to-end manner. Extensive experiments are conducted on nine real-world attributed networks and the results show that SSAGCN not only outperforms state-of-the-art approaches, but also can effectively analyzes topics associated with communities. As the future work, we will plan to extend SSAGCN to dynamic or temporal attributed networks, and conduct the evolutional analysis of communities and topics within them.



(a) Texas (b) Washington (c) Wisconsin (d) Cora (e) Citeseer (f) UAI2010 (g) Pubmed (h) Blogcatalog (i) Flickr

Fig. 8. The performance changing trends w.r.t. the number of the nearest neighbors K.

Table 5
The top 20 core members in communities. The number in the bracket denotes the membership strength.

Maarten de Rijke's community	W. Bruce Croft's community
Maarten de Rijke (0.0277)	W. Bruce Croft (0.0292)
Zhaochun Ren (0.0249)	Ameeta Agrawal (0.0211)
Pengjie Ren (0.0241)	Manos Papagelis (0.0210)
Zhumin Chen (0.0233)	Aijun An (0.0198)
Jun Ma (0.0176)	Haochen Liu (0.0197)
Xin Xin (0.0170)	Xiaobing Liu (0.0196)
Emine Yilmaz (0.0157)	Le Song (0.0196)
Edgar Meij (0.0126)	Arun Ramamurthy (0.0196)
Hongsong Li (0.0123)	Yuyu Zhang (0.0196)
Evangelos Kanoulas (0.0122)	Wenqi Fan (0.0195)
Fei Cai (0.0117)	Atanaz Babashzadeh (0.0195)
Shangsong Liang (0.0115)	Xiaorui Liu (0.0195)
Mostafa Dehghani (0.0115)	Mariam Daoud (0.0195)
Harrie Oosterhuis (0.0108)	Daxin Jiang (0.0195)
Weiwei Sun (0.0108)	Wei Jin (0.0195)
Chuan Meng (0.0107)	Mohamed Abdel Maksoud (0.0189)
Huasheng Liang (0.0106)	Qing Li (0.0189)
Dongdong Li (0.0106)	Xing Tan (0.0189)
Yujie Lin (0.0106)	Jie Zhou (0.0188)
Jingang Wang (0.0104)	Gaurav Pandey (0.0187)

CRediT authorship contribution statement

Chaobo He: Writing – original draft, Conceptualization, Methodology. **Junwei Cheng:** Software, Visualization. **Guohua Chen:** Writing – review & editing. **Quanlong Guan:** Investigation, Resources. **Xiang Fei:** Writing – review & editing. **Yong Tang:** Supervision.

Declaration of competing interest

The authors declare that they have no known competing financial interests or personal relationships that could have appeared to influence the work reported in this paper.

Data availability

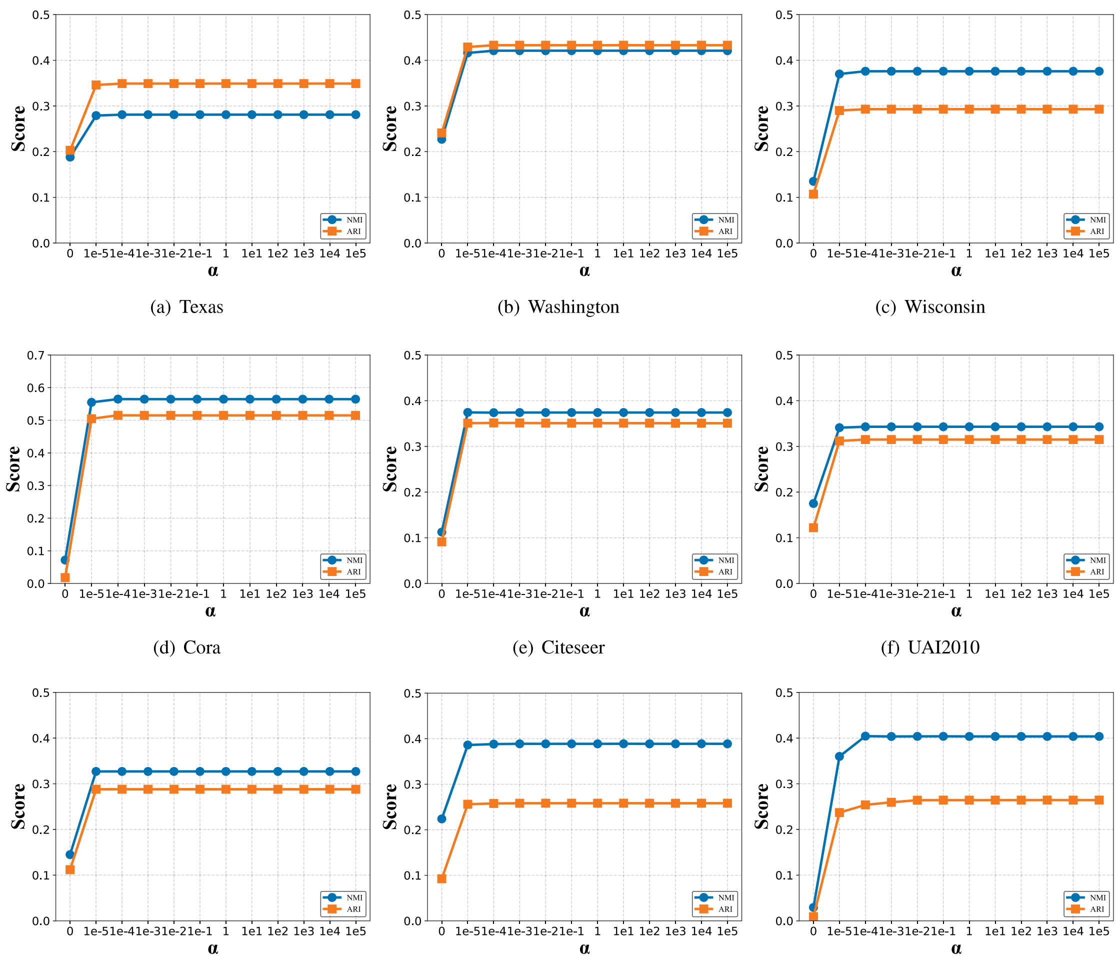
Data will be made available on request.

Acknowledgments

This work was supported by the National Natural Science Foundation of China under Grants 62077045 and 62077028, the National Key R&D Program of China under Grant 2022YFC3303604, the Science and Technology Planning Project of Guangzhou under Grant 202206030007, and the Science and Technology Planning Project of Guangdong under Grants 2020ZDZX3013 and 2023ZZ03.

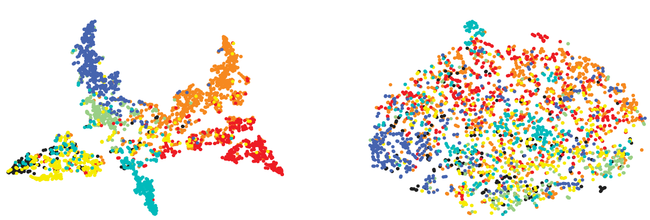
Table 6
The top 10 key attributes of the topics. The number in the bracket denotes the affiliated strength.

Maarten de Rijke's community		W. Bruce Croft's community	
t_7 (Search)	t_{20} (Ranking)	t_{19} (Recommendation)	t_{22} (Question-Answering)
Behavior (0.0313)	Learning (0.0288)	Graph (0.0402)	Conversational (0.0241)
Image (0.0292)	Efficient (0.0279)	Networks (0.0331)	Open-domain (0.0223)
Interaction (0.0271)	Model (0.0263)	Explainable (0.0308)	Features (0.0220)
Engine (0.0269)	Collections (0.0262)	Filtering (0.0259)	Summarization (0.0207)
Context-aware (0.0251)	Top-k (0.0251)	Social (0.0233)	Outside-knowledge (0.0193)
Web (0.0233)	Optimization (0.0244)	Sequential (0.0227)	Visual (0.0182)
Query (0.0231)	Evaluation (0.0221)	Collaborative (0.0218)	Embedding (0.0166)
Evaluation (0.0227)	Deep (0.0211)	Learning (0.0171)	Reasoning (0.160)
Interface (0.0217)	Neural (0.0209)	Contrastive (0.0116)	Generation (0.0151)
Cost (0.0208)	Document (0.0207)	Knowledge (0.0108)	Non-factoid (0.143)



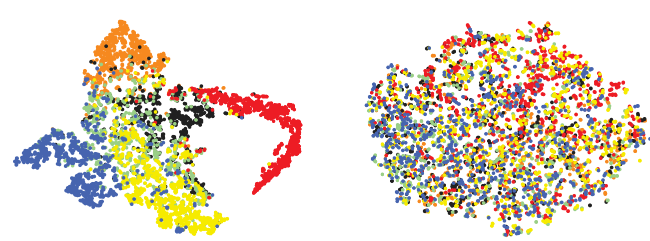
(a) Texas (b) Washington (c) Wisconsin (d) Cora (e) Citeseer (f) UAI2010 (g) Pubmed (h) Blogcatalog (i) Flickr

Fig. 9. The performance changing trends w.r.t. α.



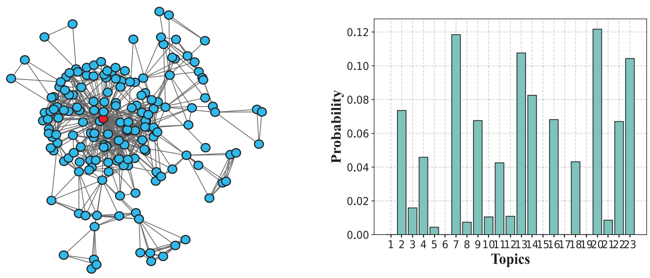
(a) $\alpha = 1$ (b) $\alpha = 0$

Fig. 10. Visualization of **H** on Cora.



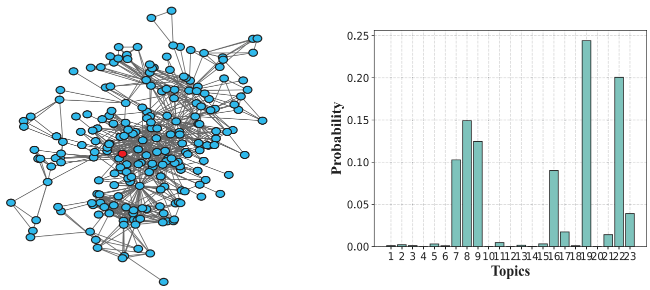
(a) $\alpha = 1$ (b) $\alpha = 0$

Fig. 11. Visualization of **H** on Blogcatalog.



(a) Community network (b) Topic distribution

Fig. 12. Maarten de Rijke's community. The red node denotes Maarten de Rijke.



(a) Community network (b) Topic distribution

Fig. 13. W. Bruce Croft's community. The red node denotes W. Bruce Croft.

References

[1] P. Chunaev, Community detection in node-attributed social networks: a survey, Comp. Sci. Rev. 37 (2020) 100286.

[2] H. Cheng, Y. Zhou, J.X. Yu, Clustering large attributed graphs: a balance between structural and attribute similarities, ACM Trans. Knowl. Discov. Data 5 (2) (2011) 12–33.

[3] X. Wang, D. Jin, X.C. Cao, L. Yang, W.X. Zhang, Semantic community identification in large attribute networks, in: Proceedings of the 30th AAAI Conference on Artificial Intelligence, AAAI, 2016, pp. 265–271.

[4] Y. Li, C.F. Sha, X. Huang, Y.C. Zhang, Community detection in attributed graphs: an embedding approach, in: Proceedings of the 32nd AAAI Conference on Artificial Intelligence, AAAI, 2018, pp. 338–345.

[5] J.H. Li, L. Huang, C.D. Wang, D. Huang, J.H. Lai, P. Chen, Attributed network embedding with micro-meso structure, ACM Trans. Knowl. Discov. Data 15 (4) (2021) 26.

[6] Z.Q. Xu, Y.P. Ke, Y. Wang, H. Cheng, J. Cheng, GBAGC: a general bayesian framework for attributed graph clustering, ACM Trans. Knowl. Discov. 9 (1) (2014) 1–43.

[7] D.X. He, H.X. Liu, Z.Y. Feng, X.B. Wang, D. Jin, W.Z. Song, Y.X. Huang, A joint community detection model: integrating directed and undirected probabilistic graphical models via factor graph with attention mechanism, IEEE Trans. Big Data 8 (4) (2022) 94–1006.

[8] A. Bojchevski, S. Günnemann, Bayesian robust attributed graph clustering: joint learning of partial anomalies and group structure, in: Proceedings of the 32nd AAAI Conference on Artificial Intelligence, AAAI, 2018, pp. 2738–2745.

[9] L. Chen, Y.J. Gao, X.R. Huang, C.S. Jensen, B.L. Zheng, Efficient distributed clustering algorithms on star-schema heterogeneous graphs, IEEE Trans. Knowl. Data Eng. 34 (10) (2022) 4781–4796.

[10] Z. Bu, H.J. Li, J. Cao, Z. Wang, G.L. Gao, Dynamic cluster formation game for attributed graph clustering, IEEE Trans. Cybern. 49 (1) (2019) 328–341.

[11] J.Y. Sun, W. Zheng, Q.F. Zhang, Z.B. Xu, Graph neural network encoding for community detection in attribute networks, IEEE Trans. Cybern. 52 (8) (2022) 7791–7804.

[12] D.Y. Bo, X. Wang, C. Shi, M.Q. Zhu, E.M. Lu, P. Cui, Structural deep clustering network, in: Proceedings of the Web Conference, WWW, 2020, pp. 1400–1410.

[13] D.X. He, Y. Song, D. Jin, Z.Y. Feng, B.B. Zhang, Z.Z. Yu, W.X. Zhang, Community-centric graph convolutional network for unsupervised community detection, in: Proceedings of the 29th International Joint Conference on Artificial Intelligence, IJCAI, 2020, pp. 3515–3521.

[14] D. Jin, Z.Z. Yu, P.F. Jiao, S.R. Pan, D.X. He, J. Wu, P.S. Yu, A survey of community detection approaches: from statistical modeling to deep learning, IEEE Trans. Knowl. Data Eng. 35 (2) (2023) 1149–1170.

[15] X. Wang, M.Q. Zhu, D.Y. Bo, P. Cui, C. Shi, J. Pei, AM-GCN: adaptive multi-channel graph convolutional networks, in: Proceedings of the 26th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, KDD, 2020, pp. 1243–1253.

[16] C.B. He, J.W. Cheng, G.H. Chen, Y. Tang, Multiple topics community detection in attributed networks, in: Proceedings of the 46th International ACM SIGIR Conference on Research and Development in Information Retrieval, SIGIR, 2023, pp. 2199–2203.

[17] K. Berahmand, M. Mohammadi, F. Saberi-Movahed, Y.F. Li, Y. Xu, Graph regularized nonnegative matrix factorization for community detection in attributed networks, IEEE Trans. Netw. Sci. Eng. 10 (1) (2023) 372–385.

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

[19] C. Wang, S.R. Pan, R.Q. Hu, G.D. Long, J. Jiang, C.Q. Zhang, Attributed graph clustering: a deep attentional embedding approach, in: Proceddings of the 28th International Joint Conference on Artificial Intelligence, IJCAI, 2019, pp. 3670–3676.

[20] C.B. He, Y.L. Zheng, J.W. Cheng, Y. Tang, G.H. Chen, H. Liu, Semi-supervised overlapping community detection in attributed graph with graph convolutional autoencoder, Inform. Sci. 608 (2022) 1464–1479.

[21] C. Wang, S.R. Pan, C.P. Yu, R.Q. Hu, G.D. Long, C.Q. Zhang, Deep neighbor-aware embedding for node clustering in attributed graphs, Pattern Recognit. 122 (2022) 108230.

[22] G.W. Du, L.H. Zhou, Z.X. Li, L.Z. Wang, K. Lü, Neighbor-aware deep multi-view clustering via graph convolutional network, Inf. Fusion 93 (2023) 330–343.

[23] M.C. Schiappa, Y.S. Rawat, M. Shah, Self-supervised learning for videos: a survey, ACM Comput. Surv. (2022) http://dx.doi.org/10.1145/3577925.

[24] L. Ericsson, H. Gouk, C.C. Loy, T.M. Hospedales, Self-supervised representation learning: introduction, advances, and challenges, IEEE Signal Process. Mag. 39 (3) (2022) 42–62.

[25] A. Baevski, W.N. Hsu, Q.T. Xu, A. Babu, J.T. Gu, M. Auli, Data2vec: a general framework for self-supervised learning in speech, vision and language, in: Proceedings of the 39th International Conference on Machine Learning, ICML, 2022, pp. 1298–1312.

[26] X. Liu, F.J. Zhang, Z.Y. Hou, L. Mian, Z.Y. Wang, J. Zhang, J. Tang, Self-supervised learning: generative or contrastive, IEEE Trans. Knowl. Data Eng. 35 (1) (2023) 857–876.

[27] Y.X. Liu, M. Jin, S.R. Pan, C. Zhou, Y. Zheng, F. Xia, P.S. Yu, Graph self-supervised learning: a survey, IEEE Trans. Knowl. Data Eng. (2022) http://dx.doi.org/10.1109/TKDE.2022.3172903.

[28] Y.C. Xie, Z. Xu, J.T. Zhang, Z.Y. Wang, S.W. Ji, Self-supervised learning of graph neural networks: a unified review, IEEE Trans. Pattern Anal. Mach. Intell. 45 (2) (2023) 2412–2429.

[29] P. Veličković, W. Fedus, W.L. Hamilton, P. Liò, Y. Bengio, R.D. Hjelm, Deep graph infomax, in: Proceedings of the 7th International Conference on Learning Representations, ICLR, 2019, pp. 1–17.

[30] Y.N. You, T.L. Chen, Y.D. Sui, T. Chen, Z.Y. Wang, Y. Shen, Graph contrastive learning with augmentations, in: Proceedings of the 34th Conference on Neural Information Processing Systems, NeurIPS, 2020, pp. 5812–5823.

[31] Y.Q. Zhu, Y.C. Xu, F. Yu, Q. Liu, S. Wu, L. Wang, Graph contrastive learning with adaptive augmentation, in: Proceedings of the Web Conference, WWW, 2021, pp. 2069–2080.

[32] N. Lee, J. Lee, C. Park, Augmentation-free self-supervised learning on graphs, in: Proceedings of the 36th AAAI Conference on Artificial Intelligence, AAAI, 2022, pp. 7372–7380.

[33] T.N. Kipf, M. Welling, Variational graph auto-encoders, in: Proceedings of the 30th Conference on Neural Information Processing Systems, NeurIPS, 2016, pp. 1–3.

[34] Z.Y. Hou, X. Liu, Y.K. Cen, Y.X. Dong, H.X. Yang, C.J. Wang, J. Tang, GraphMAE: self-supervised masked graph autoencoders, in: Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining, KDD, 2022, pp. 594–604.

[35] E. Hajiramezanali, A. Hasanzadeh, N. Duffield, K. Narayanan, M. Zhou, X. Qian, Semi-implicit graph variational autoencoders, in: Proceedings of the 33th Conference on Neural Information Processing Systems, NeurIPS, 2019, pp. 10712–10723.

[36] M.E.J. Newman, Modularity and community structure in networks, Proc. Natl. Acad. Sci. 23 (3) (2006) 8577–8582.

[37] C.B. He, X. Fei, Q.W. Cheng, H.C. Li, Z. Hu, Y. Tang, A survey of community detection in complex networks using nonnegative matrix factorization, IEEE Trans. Comput. Soc. Syst. 9 (2) (2022) 440–457.

[38] D.M. Blei, A. Ng, M. Jordan, Latent dirichlet allocation, J. Mach. Learn. Res. 3 (2003) 993–1022.

[39] M.W. Li, D.Y. Xu, J. Geng, W.C. Hong, A hybrid approach for forecasting ship motion using CNN-GRU-AM and GCWOA, Appl. Soft Comput. 114 (2022) 108084.

[40] P. Sen, G. Namata, M. Bilgic, L. Getoor, B. Galligher, T. Eliassi-Rad, Collective classification in network data, AI Mag. 29 (3) (2008) 93–106.

[41] J.D. Li, X. Hu, J.L. Tang, H. Liu, Unsupervised streaming feature selection in social media, in: Proceedings of the 24th ACM International Conference on Information and Knowledge Management, CIKM, 2015, pp. 1041–1050.

[42] T.N. Kipf, M. Welling, Semi-supervised classification with graph convolutional networks, in: Proceedings of the 5th International Conference on Learning Representations, ICLR, 2017, pp. 1–14.

[43] T. Chakraborty, A. Dalmia, A. Mukherjee, N. Ganguly, Metrics for community analysis: a survey, ACM Comput. Surv. 50 (4) (2018) 37.

[44] L.V.D. Maaten, G. Hinton, Visualizing data using t-SNE, J. Mach. Learn. Res. 86 (9) (2008) 2579–2605.

[45] V.D. Blondel, J.L. Guillaume, R. Lambiotte, E. Lefebvre, Fast unfolding of communities in large networks, J. Stat. Mech. Theory Exp. 10 (2008) 10008.