The Thirty-Second AAAI Conference
on Artificial Intelligence (AAAI-18)

Community Detection in Attributed Graphs: An Embedding Approach

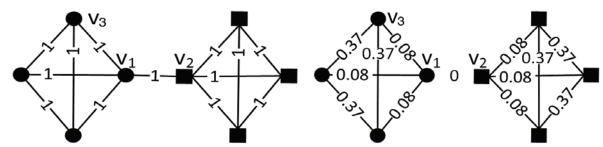
Ye Li,[1,2] Chaofeng Sha,[1,*] Xin Huang,[3,4] Yanchun Zhang[1,2,5]
[1]Shanghai Key Laboratory of Intelligence Processing, School of Computer Science, Fudan University
[2]Shanghai Key Laboratory of Data Science, School of Computer Science, Fudan University
[3]Department of Computer Science, Hong Kong Baptist University
[4]HKBU Shenzhen Institute of Research and Continuing Education
[5]Centre for Applied Informatics, College of Engineering and Science, Victoria University
{yeli15, cfsha}@fudan.edu.cn, xinhuang@comp.hkbu.edu.hk, Yanchun.Zhang@vu.edu.au
[*]Corresponding author

Abstract

Community detection is a fundamental and widely-studied problem that finds all densely-connected groups of nodes and well separates them from others in graphs. With the proliferation of rich information available for entities in real-world networks, it is useful to discover communities in attributed graphs where nodes tend to have attributes. However, most existing attributed community detection methods directly utilize the original network topology leading to poor results due to ignoring inherent community structures. In this paper, we propose a novel embedding based model to discover communities in attributed graphs. Specifically, based on the observation of densely-connected structures in communities, we develop a novel community structure embedding method to encode inherent community structures via underlying community memberships. Based on node attributes and community structure embedding, we formulate the attributed community detection as a nonnegative matrix factorization optimization problem. Moreover, we carefully design iterative updating rules to make sure of finding a converging solution. Extensive experiments conducted on 19 attributed graph datasets with overlapping and non-overlapping ground-truth communities show that our proposed model CDE can accurately identify attributed communities and significantly outperform 7 state-of-the-art methods.

Introduction

Communities are widely and naturally existed as functional modules in real-world networks, such as social networks, collaboration networks, and web graphs (Girvan and Newman 2002). Community detection (graph clustering) algorithms sever as the fundamental analysis tool for analyzing and understanding large-scale networks. In the literature, numerous studies are proposed to identify communities using network structure, including metric-based algorithms (Newman and Girvan 2004; Girvan and Newman 2002; Blondel et al. 2008; Shi and Malik 2000; Yang and Leskovec 2015) and generative models (Yang and Leskovec 2013; Wang et al. 2011; Yang et al. 2009). Besides the network topology, the entities modeled by the network nodes usually have attribute information that is important for making sense of communities. E.g., papers in citation networks have areas of keywords. Such networks with node attributes are named as *attributed graphs* (Yang, McAuley, and Leskovec 2013; Huang, Cheng, and Yu 2015).



(a) Using adjacency matrix A (b) Using community structure embedding matrix M

Figure 1: A comparison of the inherent community structures encoding performances of the adjacency matrix A and community structure embedding matrix M

Due to two different kinds of information as network structure and node attributes, it brings challenges to discover meaningful communities in attributed graphs. Several approaches that combine structural and attributed information have been proposed in (Atzmueller, Doerfel, and Mitzlaff 2016; Wang et al. 2016; Yang, McAuley, and Leskovec 2013; Huang, Cheng, and Yu 2016; Yang et al. 2009). However, in terms of unweighted networks, all of those methods directly utilize original network topology, which ignores inherent community structures by assigning each edge with the same value. Since nodes within a community are densely connected, there exist numerous densely-connected subgraphs in attributed graphs. Intuitively, edges that form those densely-connected subgraphs are much more likely to construct a corresponding community than edges that connect separate subgraphs. Consider the example illustrated in Fig.1(a), there are two linked node pairs (v_1, v_3) and (v_1, v_2), where v_1 and v_3 belong to the same community while v_1 and v_2 do not. The original network topology doesn't reflect the discrepancy of those two pairs, because both pairs are assigned with the same value, 1. However, (v_1, v_3) is more important than (v_1, v_2) in constructing a community obviously. So utilizing original network topology directly causes indiscriminately penalizing node pairs whether in densely-connected structures or not (Zhang, King, and Lyu 2015). While nodes in a community are densely-connected, they should also share homogenous node attributes. As one of the state-of-the-art methods, SCI (Wang et al. 2016) combines original network topology and node attributes with a unified objective function. However, SCI doesn't factorize the sparse node attributes matrix but treats it as the basis to fit the community membership (refer to equation (2) of (Wang et al. 2016)), which could degrade the effectiveness of SCI due to the redundancy and noise in node attribute matrix.

In this paper, we study the community detection problem in attributed graphs, that is, to find all meaningful communities such that nodes are densely-connected and have homogeneous attributes in the same community. Communities are allowed to overlap, i.e., one node can present in multiple communities. To address the problem of directly using original network topology and leverage the node attributes much more thoroughly, we propose a novel community structure embedding based model for community detection incorporating community structures and node attributes. Specifically, we propose a novel community structure embedding method, based on the observation of densely-connected structure in the community, to encode inherent community structures via underlying community memberships. Then, leveraging on the information of node attributes and the community structure embedding, we formulate the attributed community detection as a nonnegative matrix factorization optimization problem and carefully design updating rules to ensure finding a converging solution.

Our main contributions are summarized as follows.

- We propose a novel community structure embedding method to encode inherent community structures for community detection purpose via underlying community memberships.
- We leverage the information of node attributes and explore associated attributes for detected communities. Furthermore, we integrate community structure embedding matrix and node attributes matrix, and formulate our *Community Detection in attributed graphs: an Embedding approach* (CDE) model as an optimization problem to identify all communities in attributed graphs.
- We carefully design iterative updating rules and theoretically prove the convergence of the proposed method. Extensive experiments on 19 real-world attributed graph datasets show that our proposed CDE significantly outperforms 7 state-of-the-art methods.

Related Work

Works related to this paper include structural graph clustering, attributed graph clustering, and network embedding.

Structural Graph Clustering. Graph clustering (Community detection) algorithms are widely studied. We focus on generative model based algorithms in this paper, which assume that edges are generated with the probabilities depending on their community memberships. They design probability models to fit network structures, like (Yang and Leskovec 2013; Wang et al. 2011; Yang et al. 2009). Another stream of related work for structural graph clustering utilizes the technique of nonnegative matrix factorization (NMF), like (Wang et al. 2008; Yang et al. 2012; Wang et al. 2016; Yang et al. 2012; Wang et al. 2011). Those algorithms are based on the principle that the adjacency matrix can be factorized into the linear combination of community membership matrix. And, (Yang et al. 2012) factorizes the smoothed version of adjacency matrix obtained by using random walk.However, all of those methods directly utilize original network topology that ignores the inherent community structures.

Attributed Graph Clustering. In the literature, there exist several studies on attributed graph clustering (Atzmueller, Doerfel, and Mitzlaff 2016; Wang et al. 2016; Yang, McAuley, and Leskovec 2013; Huang, Cheng, and Yu 2016; Yang et al. 2009; He et al. 2017; Huang, Cheng, and Yu 2015).As one representative method, (Yang et al. 2009) proposes a discriminative model to incorporate the node attributes into the network structure model by choosing weight vectors. Besides constructing generative models, some other approaches use NMF techniques to combine network structure and node attributes. One of the state-of-the-art algorithms is SCI (Wang et al. 2016) which uses NMF technique to combine observed network structure and node attributes. However, SCI directly factorizes the adjacency matrix and doesn't focus on factorizing node attributes matrix, which overlooks the various significances of edges in forming community structures and causes failure to leverage the node attributes thoroughly.

Different from all these structural graph clustering and attributed graph clustering methods, we develop a novel community structure embedding matrix to encode inherent community structures instead of simply using the observed network topology for community detection. In addition, our work focuses on factorizing both community structure embedding matrix and node attributes matrix simultaneously to yield better performance for detecting communities in attributed graphs.

Network Embedding. Our work relates to network embedding that learns latent vectors for representing nodes, like (Perozzi, Al-Rfou, and Skiena 2014; Cao, Lu, and Xu 2015; Grover and Leskovec 2016; Zhang et al. 2016). One of representative works, DeepWalk (Perozzi, Al-Rfou, and Skiena 2014), learns latent representations of nodes by treating random walks as the equivalence of sentences. In addition, (Wang et al. 2017) considers community structure in learning node representations. Different from existing network embedding methods, our work encodes inherent community structures for community detection purpose via underlying community memberships.

Preliminaries

Problem Statement. Consider an undirected and weighted attributed graph $G = (V, E, W, T)$, where $V = \{v_1, v_2, ..., v_n\}$ is a set of n nodes, edges set $E \subseteq V \times V$, W_{ij} represents the weight value of edge (v_i, v_j), and $T \in \{0,1\}_{n\times s}$ is node attributes matrix with i-th row represents the attribute value of node v_i using an s-dimensional binary-valued vector. In addition, we use the adjacency matrix $A \in \{0,1\}_{n\times n}$ to represent the observed graph structure, i.e., if an edge $(v_i, v_j) \in E$, $A_{ij} = 1$; Otherwise, $A_{ij} = 0$.

Note that throughout this paper, we restrict our discussions on undirected and unweighted attributed graphs, however, our method can be easily extended to handle directed and weighted attributed graphs. In the unweighted graph G, for any pair of nodes v_i and v_j, the weight $W_{ij} = A_{ij} \in \{0, 1\}$.

Given an attributed graph G and the number of communities K, the community detection problem studied in this paper is to find K groups of nodes $\mathcal{G} = \{g_1, \dots, g_K\}$ such that: (1) in terms of structure (i.e., edge connections in G), the nodes within groups are densely connected, while the nodes in different groups are sparsely connected; and (2) in terms of attribute (i.e., attribute values on the nodes in G), the nodes within groups have homogeneous attribute values, while the nodes in different communities may have diverse attribute values. We note that the communities studied in this paper are allowed to overlap, i.e., $g_i \cap g_j \neq \emptyset$ may exist.

Nonnegative Matrix Factorization. Given n nodes and the number of communities K, we define the community membership matrix $U_{n\times K}$ where each U_{ij} presents the tendency that i-th node belongs to j-th community for $1 \leq i \leq n$ and $1 \leq j \leq K$; $U_{i:}$ presents i-th row of U. The expected number of edges between pairs of nodes represented by UU^T should approximate the visible network structure represented by the adjacency matrix A. Thus, based on minimizing reconstruction cost, the problem of community detection (*SNMF* (Wang et al. 2011)) can be formulated as follows.

$$\min_{U\geq 0} ||A - UU^T||_F^2. \tag{1}$$

SNMF using Eq.1 has an obvious limitation, it only utilizes the observed network topology by directly factorizing the adjacency matrix A. For every linked node pairs v_i and v_j, $A_{ij} = 1$ whether they belong to the same community or not. Thus, the inherent community structures cannot be represented by A. To solve this drawback, we design a novel community structure embedding method in the next section.

Community Structure Embedding Method

In this section, we proposed a novel nonnegative matrix factorization (NMF) based approach for community detection, which uses our novel community structure embedding method to encode inherent community structures for community detection purpose.

Community Structure Embedding

To offer a good depiction of inherent community structures in graphs, we propose a novel community structure embedding method to quantify the structural closeness of nodes, according to their potential community membership similarities. Specifically, we design function $\mathcal{F}$ to measure community membership similarity. Then, based on the similarity measurement, we adopt skip-gram with negative-sampling (SGNS) (Mikolov et al. 2013) to explore the network structure and depict the underlying community structures. Finally, we obtain a community structure embedding matrix that encodes the inherent community structures.

Community Membership Similarity. We start with a concise and reasonable observation that each linked pair of nodes should have a certain tendency to fall into the same community because nodes are densely connected in a community. Consider two nodes v_i and v_j in graph G and the product of community memberships $U_{i:}U_{j:}^T \geq 0$, we design our similarity measurement function $\mathcal{F}(i,j) \in [0, 1)$ using the sigmoid function σ to measure the similarity of their community memberships as follows:

$$\mathcal{F}(i,j) = 2\sigma(U_{i:}U_{j:}^T) - 1 = 2 \times (\frac{1}{1 + e^{-U_{i:}U_{j:}^T}}) - 1 \tag{2}$$

The choice of the sigmoid function σ is mainly because it is a bounded differentiable real function for all real input values. Since $\mathcal{F}(i,j)$ is the linear transformation of $\sigma(U_{i:}U_{j:}^T)$, we focus on the term $\sigma(U_{i:}U_{j:}^T)$ in remaining discussion.

Community Structure Embedding Matrix (M). Our community structure embedding method tries to maximize $\sigma(U_{i:}U_{j:}^T)$ for connected nodes v_i and v_j, meanwhile minimizing $\sigma(U_{i:}U_{j:}^T)$ for a pair of randomly selected nodes v_i and v_j. In the real-world life, most large networks are very sparse and a pair of randomly selected nodes is likely to be connected with a low probability. Thus, based on the skip-gram with negative-sampling (SGNS) (Mikolov et al. 2013), we formulate the negative sampling objective function for nodes v_i and v_j as follows:

$$\ell(i,j) = W_{ij}(\log\sigma(U_{i:}U_{j:}^T) + \kappa\mathbb{E}_{j_N\sim P_V}[\log\sigma(-U_{i:}U_{j_N:}^T)]) \tag{3}$$

where κ is the number of negative samples, $\log \stackrel{def}{=} \log_e$, and j_N is the randomly sampled node with the empirical unigram distribution $P_V(i) = \frac{d_i}{D}$. $d_i = \sum_j W_{ij}$ is the degree of node i and $D = \sum_i d_i$ is the total degree for graph G. Moreover, we can explicitly express the term $\mathbb{E}_{j_N\sim P_V}$ and rewrite Eq.3 as follows.

$$\ell(i,j) = W_{ij}\log\sigma(U_{i:}U_{j:}^T) + \kappa\frac{d_i d_j}{D}\log\sigma(-U_{i:}U_{j:}^T) \tag{4}$$

To optimize Eq.4, we can find Eq.4 partial derivative with respect to variable $U_{i:}U_{j:}^T$ as:

$$\frac{\partial\ell(i,j)}{\partial(U_{i:}U_{j:}^T)} = W_{ij}\sigma(-(U_{i:}U_{j:}^T)) - \kappa\frac{d_i d_j}{D}\sigma(U_{i:}U_{j:}^T) \tag{5}$$

Finally, we obtain the optimal community structure embedding for arbitrary pair of nodes by comparing the derivative in Eq.5 to 0, and reach that as:

$$U_{i:}U_{j:}^T = \log\frac{W_{ij}D}{d_i d_j} - \log\kappa \tag{6}$$

The Eq.6 could be negative. It suggests that we should shift those negative results to 0 by selecting the maximum value of $U_{i:}U_{j:}^T$ and 0.

Overall, based on the above analysis results, we define a new community structure embedding matrix $M_{n\times n} \in \mathcal{R}_{n\times n}$ with the following formulation:

$$M_{ij} = max\{U_{i:}U_{j:}^T, 0\} = max\{\log\frac{W_{ij}D}{d_i d_j} - \log\kappa, 0\} \tag{7}$$

For disconnected pairs of nodes v_i and v_j with $W_{ij} = 0$, we set the corresponding $M_{ij} = 0$.

The community structure embedding matrix M is able to encode latent densely-connected subgraphs and explore inherent community structures. In addition, it can be easily extended to all kinds of networks (directed/undirected and weighted/unweighted networks), which shows the wide applications of our community structure embedding method.

Example. We present a toy example to demonstrate the superiority of our community structure embedding matrix M ($\kappa = 2$) in encoding proper inherent community structures within unweighted graphs.

Fig.1(a) illustrates an unweighted graph with two communities (marked by different shapes). Since the adjacency matrix sets the same value for each edge (1 in Fig.1(a)), it cannot determine which edges are more important when detecting two communities. Whereas our community structure embedding matrix can assign more weights to the connections within each community while assigning less weight to the connection between two communities, as is shown in Fig.1(b). Specifically, our community structure embedding method reduces the weights of connections between nodes with high degrees, for they are more likely to play key roles in dividing communities. For example, in Fig.1(a), both nodes v_1 and v_2 have the maximum degree. Therefore, the edge (v_1, v_2) is eliminated in Fig.1(b) ($M_{v_1 v_2} = 0$).

Structure Embedding based Optimization.

Unlike previous NMF-based studies (e.g. *SNMF*) that directly utilize visible graph structure by the adjacency matrix A, we use the community structure embedding matrix M to depict inherent community structures in graphs here.

The problem of community detection using community structure embedding matrix M without the consideration of attributes can be formulated as

$$\min_{U \geq 0} \mathcal{L}(U) = ||M - UU^T||_F^2. \tag{8}$$

CDE Model

In this section, we propose CDE model to address the problem of community detection on attributed graphs using structural information and node attributes.

Community Attributes.

Recall that in the attributed graphs, the communities should not only have densely-connected structure but also have homogeneous attributes. Besides, different communities prefer different attributes. To model these, we define the community-attribute matrix $C \in R_{K \times s}$ where C_{ir} represents the preference of i-th community for r-th dimension of node attributes.

We formulate the discovery of community membership and community attributes as a nonnegative matrix factorization problem below, which optimizes both community membership matrix U and community-attribute matrix C by factorizing node attributes matrix T.

$$\min_{U \geq 0, C \geq 0} \mathcal{L}(C) = ||T - UC||_F^2 + \alpha \sum_i ||C_{:i}||_1^2. \tag{9}$$

where α is a nonnegative parameter to control the sparsity of C.

Since different communities tend to prefer different attributes, and even worse, some attributes themselves show mutual exclusive. To address this problem, we involve the l_1 norm sparsity to each column of C, which reduces interference from unimportant node attributes for each community.

Unified Objective Function for CDE

In the attributed graph G, the goal of CDE is to find K communities such that nodes within communities are densely connected and have homogeneous attribute values. By incorporating the objective functions for community structures embedding and node attributes respectively as Eq.8 and Eq.9, we define the unified objective function for our CDE model as follows.

$$\min_{U \geq 0, C \geq 0} \mathcal{L}(U, C) = ||T - UC||_F^2 + \alpha \sum_i ||C_{:i}||_1^2 + \beta ||M - UU^T||_F^2 \tag{10}$$

where β is a positive parameter to balance the contributions of node attributes matrix T and community structure embedding matrix M. Specifically, bigger β leads to more reliance on the factorization of community structure embedding matrix M for determining communities.

Identify Communities by U. For node v_i, we define a set of communities containing v_i as $\psi(i) \subseteq \{1, \ldots, K\}$. After solving the optimization problem in Eq.10, we obtain the optimal community membership matrix U. To identify non-overlapping communities, we find exactly one community that v_i achieves the maximum value of $U_{i:}$, i.e., $\psi(i) = \arg\max_{1 \leq j \leq K} U_{ij}$. As for detecting overlapping communities, we identify that node v_i belongs to the j-th community when U_{ij} is higher than a predefined threshold ε, i.e., $\psi(i) = \{1 \leq j \leq K | U_{ij} > \varepsilon\}$. Following (Zhang, King, and Lyu 2015), we set $\varepsilon = 0.1$ in our experiments.

Iteratively Updating Rules for CDE

As the Eq.10 is not convex, we provide an iteratively updating rules for CDE, which are based on Majorization-Minimization framework in (Hunter and Lange 2004). For each iteration, we update U with C fixed and then C with U fixed.

For updating U with C fixed, we extract terms related to U in Eq.10 as follows:

$$\min_{U \geq 0} \mathcal{L}(U) = \beta ||M - UU^T||_F^2 + ||T - UC||_F^2. \tag{11}$$

After the initialization of U, we can use following updating rule for U with guarantee of convergence:

$$U_{ij} \leftarrow U_{ij} \left(\frac{(TC^T - UCC^T + 2\beta MU)_{ij}}{(2\beta UU^T U)_{ij}} \right)^{\frac{1}{4}} \tag{12}$$

When it comes to updating C with U fixed, we have to solve the following optimization problem extracted from Eq.10:

$$\min_{C \geq 0} \mathcal{L}(C) = ||T - UC||_F^2 + \alpha \sum_i ||C_{:i}||_1^2. \tag{13}$$

According to (Kim and Park 2008), this optimization problem equals to following formulation:

$$\min_{C\geq 0}\mathcal{L}(C)=||\begin{pmatrix}U\\ \sqrt{\alpha}\vec{e}\end{pmatrix}C-\begin{pmatrix}T\\ \vec{0}\end{pmatrix}||_F^2 \quad (14)$$

Where $\vec{e}$ is a row vector with the same length of rows in U, in which all elements are equal to 1. And $\vec{0}$ is a 0 vector with the same length of rows in T. We define $U'=\begin{pmatrix}U\\ \sqrt{\alpha}\vec{e}\end{pmatrix}$ and $T'=\begin{pmatrix}T\\ \vec{0}\end{pmatrix}$, then we write the updating rule for C in Eq.14 as follows.

$$C_{ij}\leftarrow C_{ij}\frac{(U'^TT')_{ij}}{(U'^TU'C)_{ij}} \quad (15)$$

For Eq.12, the complexity depends on the term MU, which is $O(n^2K)$. When it comes to Eq.15, the complexity relies on U'^TC' which is $O(nsK)$ with the condition that $n\leq s$ or $O(n^2K)$ otherwise. So the complexity for each iteration is $O(nsK)$ when $n\leq s$ or $O(n^2K)$ otherwise.

Since the embedding matrix M, the node attributes matrix T and the community-attribute matrix C are very sparse for large attributed graphs, e.g. Flickr (Ruan, Fuhry, and Parthasarathy 2013), we speed up our iteratively updating rules (Eq.12, Eq.15) by only storing positive elements and parallelizing matrix multiplication using 64 processors on our server.

Proof of Convergence. We now prove the convergence of updating rule for U using the auxiliary function approach and updating rule for C respectively.

Before proving the convergence of updating rule for U, we introduce the definition of auxiliary function and Lemma 1 in (Lee and Seung 2001).

Definition 1 (Lee and Seung 2001) $\tilde{U}\in R_{n\times K}$ is an auxiliary matrix for U, function $\mathcal{Q}(U,\tilde{U})$ is an auxiliary function of $\mathcal{L}(U)$ if $\mathcal{Q}(U,\tilde{U})\geq\mathcal{L}(U)$ and $\mathcal{Q}(U,U)=\mathcal{L}(U)$ for any $U,\tilde{U}$.

Lemma 1 (Lee and Seung 2001) If $\mathcal{Q}(U,\tilde{U})$ is an auxiliary function of $\mathcal{L}(U)$, $\mathcal{L}(U)$ is non-increasing under the updating rule $U^{t+1}=argmin_U\mathcal{Q}(U,U^t)$.

To design an auxiliary function $\mathcal{Q}(U,\tilde{U})$ for Eq.12, we rewrite it as follows:

$$\begin{aligned}\mathcal{L}(U)=&tr(TT^T-UCT^T-TC^TU^T+UCC^TU^T)\\ &+\beta tr(MM^T-2MUU^T+UU^TUU^T) \quad (16)\end{aligned}$$

Derived from Lemma 5 and 6 (Wang et al. 2011), we obtain the following upper bounds for two positive terms in Eq.16:

$$tr(UU^TUU^T)\leq tr(P\tilde{U}^T\tilde{U})\leq tr(R\tilde{U}^T\tilde{U}\tilde{U}^T)$$
$$tr(UCC^TU^T)\leq\frac{1}{2}tr(CC^TY^T\tilde{U}+CC^T\tilde{U}^TY)$$

Where $P_{ij}=\frac{(U^TU)^2_{ij}}{(\tilde{U}^T\tilde{U})_{ij}}$, $R_{ij}=\frac{U^4_{ij}}{\tilde{U}^3_{ij}}$ and $Y_{ij}=\frac{U^2_{ij}}{\tilde{U}_{ij}}$.

For negative terms in Eq.16 that involve U, we use Lemma 2 and 3 (Wang et al. 2011) to derive the lower bounds as follows:

$$tr(MUU^T)\geq tr(\tilde{U}^TMZ)+tr(Z^TM\tilde{U})+tr(\tilde{U}^TM\tilde{U})$$
$$tr(TC^TU^T)=tr(UCT^T)\geq tr(CT^TZ)+tr(CT^T\tilde{U})$$

Where $Z_{ij}=\tilde{U}_{ij}log(\frac{U_{ij}}{\tilde{U}_{ij}})$. Then we construct our auxiliary function $\mathcal{Q}(U,\tilde{U})$ for Eq.12:

$$\begin{aligned}\mathcal{Q}(U,\tilde{U})\stackrel{def}{=}&tr(TT^T+\beta MM^T)+\beta tr(R\tilde{U}^T\tilde{U}\tilde{U}^T)\\ &-2tr(CT^TZ)-2tr(CT^T\tilde{U})-2\beta tr(\tilde{U}^TMZ)\\ &-2\beta tr(Z^TM\tilde{U})-2\beta tr(\tilde{U}^TM\tilde{U})\\ &+\frac{1}{2}tr(CC^TY^T\tilde{U}+CC^T\tilde{U}^TY) \quad (17)\end{aligned}$$

Applying the KKT condition to our constructed $\mathcal{Q}(U,\tilde{U})$, we obtain the updating rule for U (Eq.12), which could guarantee convergence.

As for the convergence of the updating rule for C (Eq.15), the proof is provided in (Kim and Park 2008).

Thus, for each iteration, our iteratively updating rules guarantee no increase for Eq.10.

Experiments

In this section, we performed extensive experiments to evaluate the effectiveness of our proposed CDE on 19 real graph datasets with ground-truth communities. We believe that it is hard to compare the quality of community results when the numbers of communities are different for baseline methods. Thus, we set the number of detected communities K as the number of ground-truth communities. Moreover, we conducted our CDE on each dataset with 10 different initializations and reported the average of 10 results.

Our algorithms are implemented in Matlab and C++, and all experiments are conducted on a Windows Servers with Xeon 64-core CPU (2.70 GHz) and 128G main memory.

Datasets

We use 19 attributed graph datasets with ground-truth communities for evaluation in our experiments. The network statistics are reported in Table 1.

For *Non-Overlapping* Ground-truth Communities, we use 6 datasets of Citeseer, Cora, Cornell, Texas, Washington, and Wisconsin, which are available at the website[1].

When it comes to *Overlapping* Ground-truth Communities, we use graph datasets from 3 different domains, Philosophers network (Ahn, Bagrow, and Lehmann 2010), Flickr (Ruan, Fuhry, and Parthasarathy 2013), and Facebook. The Facebook dataset is a set of Facebook ego-networks. It contains 10 different ego-networks with manually identified circles (Leskovec and Mcauley 2012). The information of user profiles is treated as node attributes, and social circles formed by friends are regarded as ground-truth communities. In addition, we merge these 10 Facebook ego-networks into one entire Facebook network with 4039 nodes and 88234 edges.

[1] http://linqs.cs.umd.edu/projects/projects/lbc/

Dataset	$\lvert V \rvert$	$\lvert E \rvert$	s	K	AS	AN
Non-overlapping						
Cornell	195	283	1703	5	39	1
Texas	187	280	1703	5	37.4	1
Washington	230	366	1703	5	46	1
Wisconsin	265	459	1703	5	53	1
Cora	2708	5278	1433	7	386.86	1
Citeseer	3312	4536	3703	6	552	1
Overlapping						
FB ego-network 0	348	2852	224	24	13.54	0.93
FB ego-network 107	1046	27783	576	9	55.67	0.48
FB ego-network 1684	793	14810	319	17	45.71	0.98
FB ego-network 1912	756	30772	480	46	23.15	1.41
FB ego-network 3437	548	5347	262	32	6	0.35
FB ego-network 348	228	3416	161	14	40.5	2.49
FB ego-network 3980	60	198	42	17	3.41	0.97
FB ego-network 414	160	1843	105	7	25.43	1.11
FB ego-network 686	171	1824	63	14	34.64	2.84
FB ego-network 698	67	331	48	13	6.54	1.27
Facebook	4039	88234	10	193	21.93	1.03
Philosophers	1218	5972	5770	1220	10.97	10.99
Flickr	16710	716063	1156	5436	273.10	88.84

Table 1: Dataset Statistics. $|V|$: number of nodes, $|E|$: number of edges, s: number of node attributes, K: number of communities, AS: average size of communities, AN: average community memberships, FB: Facebook dataset.

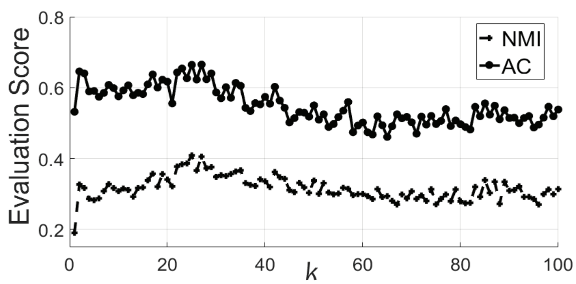
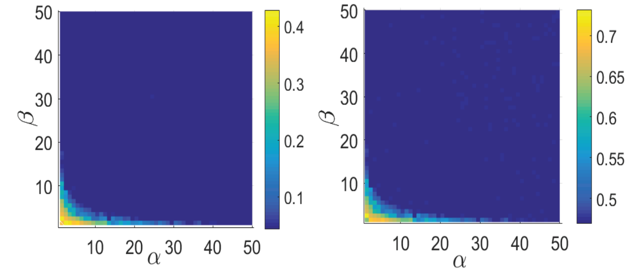


Figure 2: Fixing $\alpha = \beta = 1$ then varying κ from 1 to 100 on Wisconsin dataset

Parameter Sensitivity Analysis

In this section, we perform parameter sensitivity analysis of CDE on Wisconsin dataset. Similar results can also be found using other datasets. CDE has three parameters: α is a non-negative constant that controls the sparsity of community-attribute matrix C, β is a positive constant to balance the contributions of node attributes and community structure embedding, and κ determines the number of negative samples for community structure embedding method. We first set $\alpha = \beta = 1$ to treat node attributes and community structure embedding with the same importance. Then, we test CDE on Wisconsin by varying parameter κ from 1 to 100. Fig.2 shows CDE achieves better performances in the range of $\kappa = 22$ through $\kappa = 29$. More specifically, CDE achieves the maximum scores of $AC = 0.6645$ and $NMI = 0.409$ when $\kappa = 25$, which suggests that a suitable number of sampling nodes could improve the performance of CDE.

In terms of the parameters α and β, we set $\kappa = 25$ and vary α and β from 1 to 50 respectively. Figure 3 shows the



(a) NMI (b) AC

Figure 3: Fixing $\kappa = 25$ then varying α and β from 1 to 50 respectively on Wisconsin dataset

corresponding results on Wisconsin dataset, in which CDE achieves maximum scores of $AC = 0.7321$ and $NMI = 0.4284$ when $\alpha = 1$ and $\beta = 2$. Those results indicate our community structure embedding method could extract essential structural information in separating different communities from the original network topology, and further demonstrate the superiority of our community structure embedding method in encoding inherent community structures.

Evaluation on Overlapping Communities

In this experiment, we evaluate the effectiveness of our proposed CDE on attributed graphs with overlapping ground-truth communities. We use 13 datasets including 10 Facebook ego-networks, Facebook network, Philosophers, and Flickr.

We compare CDE with 4 state-of-the-art methods:*Bigclam* (Yang and Leskovec 2013), *Circles* (Leskovec and Mcauley 2012), *CESNA* (Yang, McAuley, and Leskovec 2013) and *SCI* (Wang et al. 2016).

Note that all comparison methods make use of network structure with node attributes as well, except for *Bigclam* (Yang and Leskovec 2013) that only uses network structures. Since *Bigclam* (Yang and Leskovec 2013) achieved better performances among other baseline algorithms on several datasets, we also chose it as one of our baseline algorithms.

For all baseline methods, we use the implementations provided by their authors and set their parameters by default. Both CDE and *SCI* (Wang et al. 2016) set the threshold ε to be 0.1 for identifying community membership. In addition, our method CDE sets the parameters $\alpha = 1, \beta = 2$ and $\kappa = 5$.

To evaluate the quality of discovered communities, we use two evaluation metrics of *F1-score* and *Jaccard Similarity* respectively to reflect the alignment between discovered communities and ground-truth communities. We adopt the same evaluation procedure used in (Yang and Leskovec 2013) that every detected community is matched with its most similar ground-truth community. Given a set of discovered communities C and a set of ground-truth communities C^*, a unified formulation of F1-score and Jaccard Similarity is defined as:

	F1-Score					Jaccard Similarity				
Dataset	Bigclam	CESNA	Circles	SCI	CDE	Bigclam	CESNA	Circles	SCI	CDE
FB ego-network 0	0.2632	0.2638	0.2845	0.2104	**0.3190**	0.1617	0.1635	0.1844	0.1255	**0.2022**
FB ego-network 107	0.3593	0.3526	0.2722	0.1932	**0.3755**	0.2589	0.2512	0.1755	0.1203	**0.2677**
FB ego-network 1684	0.3652	0.3850	0.3022	0.2290	**0.5798**	0.2655	0.2656	0.1947	0.1405	**0.4457**
FB ego-network 1912	0.3542	0.3506	0.2694	0.2787	**0.3798**	0.2443	0.2417	0.1744	0.1872	**0.2694**
FB ego-network 3437	0.2109	0.2125	0.1004	0.1909	**0.2191**	0.1226	**0.1311**	0.0545	0.1130	0.1272
FB ego-network 348	0.4917	0.4943	0.5209	0.4469	**0.5389**	0.3649	0.3684	0.3937	0.3068	**0.4092**
FB ego-network 3980	0.4468	0.4312	0.3277	0.3502	**0.5040**	0.3147	0.3041	0.2127	0.2336	**0.3967**
FB ego-network 414	0.5703	0.6181	0.5090	0.5617	**0.6531**	0.4446	0.4878	0.3670	0.4270	**0.5392**
FB ego-network 686	0.3583	0.3638	0.5242	0.4340	**0.5252**	0.2272	0.2372	0.3828	0.2915	**0.3679**
FB ego-network 698	0.5276	0.5222	0.3715	0.4323	**0.5744**	0.3898	0.3811	0.2400	0.2942	**0.4614**
Facebook	0.2867	0.3277	NA	0.0770	**0.3609**	0.1977	0.2265	NA	0.0413	**0.2502**
Philosophers	0.3871	0.3929	NA	0.3313	**0.4217**	0.2478	0.2544	NA	0.2244	**0.2813**
Flickr	0.0833	0.1014	NA	0.0512	**0.1326**	0.0445	0.0543	NA	0.0231	**0.0723**

Table 2: Quality evaluation (in terms of F1-Score and Jaccard Similarity) on networks with overlapping ground-truth communities. NA means that the task is not completed in limited time.

	AC					NMI				
Dataset	SNMF	NC	PCL-DC	SCI	CDE	SNMF	NC	PCL-DC	SCI	CDE
Cornell	0.3692	0.3538	0.3512	0.4769	**0.6154**	0.0762	0.0855	0.0873	0.1516	**0.3403**
Texas	0.4019	0.4545	0.3850	0.6096	**0.6150**	0.1022	0.0706	0.0729	0.2153	**0.3208**
Washington	0.3009	0.4348	0.4608	0.5173	**0.6696**	0.0321	0.0591	0.1195	0.1304	**0.4079**
Wisconsin	0.3773	0.3170	0.3773	0.5283	**0.7321**	0.0842	0.0507	0.0778	0.1823	**0.4284**
Cora	0.4323	0.2622	0.5823	0.4121	**0.6555**	0.2996	0.1731	0.4071	0.2138	**0.5037**
Citeseer	0.3079	0.4094	0.4682	0.3260	**0.5827**	0.1044	0.1998	0.2246	0.0758	**0.2985**

Table 3: Quality evaluation (in terms of AC and NMI) on networks with non-overlapping ground-truth communities.

$$\frac{1}{2|C^*|}\sum_{C_i^* \in C^*}\max_{C_j \in C}\delta(C_i^*, C_j) + \frac{1}{2|C|}\sum_{C_j \in C}\max_{C_i^* \in C^*}\delta(C_i^*, C_j).$$

When $\delta(C_i^*, C_j)$ is defined as the harmonic mean of C_i^* and C_j, it is the F1-score metric; On the other hand, when $\delta(C_i^*, C_j) = \frac{|C_i^* \cap C_j|}{|C_i^* \cup C_j|}$, it is the Jaccard Similarity metric. Both metrics with larger values are better.

We report the F1-score and Jaccard Similarity of all methods in Table 2. The results indicate that CDE outperforms all comparison algorithms in the overlapping community detection task, except one case of facebook ego-network 3437 of Facebook measured by Jaccard Similarity. Since *Circles* (Leskovec and Mcauley 2012) cannot handle networks with more than 1200 nodes in a reasonable time, we only test it on 10 small Facebook ego-networks.

Moreover, the running time of CDE on Flickr is less than baseline methods' except for *Bigclam*. However, CDE outperforms *Bigclam* in the quality evaluation on all datasets in Table 2.

Evaluation on Non-Overlapping Communities

To evaluate the accuracy of identified non-overlapping communities, we compare CDE with 4 state-of-the-art community detection algorithms. We implemented *SNMF* (Wang et al. 2011) and used source codes of *Normalized Cut(NC)* (Shi and Malik 2000), *PCL-DC* (Yang et al. 2009) and *SCI* (Wang et al. 2016) provided by authors.

To measure the accuracy of identified communities, we use the metrics of Accuracy (AC), which measures the percentage of correct community memberships obtained (Wu and Liu 2010), and Normalized Mutual Information (NMI). Note that each node is assigned to one community.

We compare CDE with four other baseline methods using the 6 networks, Cornell, Texas, Washington, Wisconsin, Cora, and Citeseer, with non-overlapping ground-truth communities. In our experiments, we set $\alpha = 1, \beta = 2$ and varied κ from 1 to 30 to get the maximum AC and NMI. In addition, the running time of CDE for each experiment is less than 40 seconds. The results reported in Table 3 show that CDE clearly outperforms 4 other baseline methods on all datasets with significant improvements. Furthermore, the results of comparison algorithms on the same datasets are also available in their papers, which once-again demonstrates the performance improvements achieved by CDE.

Conclusion

In this paper, we studied the problem of community detection in attributed graphs. Specifically, we proposed a novel community structure embedding method to encode inherent community structures for community detection purpose and showed its superiority in separating different communities in contrast with original network topology. Furthermore, we learned the associated attributes for underlying communities from the given node attributes. Based on the community structure embedding and node attributes matrix, we formulated our CDE model as a nonnegative matrix factorization optimization problem. Finally, extensive experiments on 19 real-world attributed graph datasets showed that our CDE model can effectively discover the overlapping and non-overlapping communities, significantly outperforming 7 state-of-the-art baseline methods.

Acknowledgments

The authors would like to thank the anonymous reviewers for their insightful comments. This work was supported by NSFC (Nos. 61332013, 61572135, 61702435) and Hong Kong General Research Fund (GRF) Project No. HKBU 12200917.

References

Ahn, Y.-Y.; Bagrow, J. P.; and Lehmann, S. 2010. Link communities reveal multiscale complexity in networks. *Nature* 466(7307):761–764.

Atzmueller, M.; Doerfel, S.; and Mitzlaff, F. 2016. Description-oriented community detection using exhaustive subgroup discovery. *Information Sciences* 329:965–984.

Blondel, V. D.; Guillaume, J.-L.; Lambiotte, R.; and Lefebvre, E. 2008. Fast unfolding of communities in large networks. *Journal of statistical mechanics: theory and experiment* 2008(10):P10008.

Cao, S.; Lu, W.; and Xu, Q. 2015. Grarep: Learning graph representations with global structural information. In *Proceedings of the 24th ACM International on Conference on Information and Knowledge Management*, 891–900. ACM.

Girvan, M., and Newman, M. E. 2002. Community structure in social and biological networks. *Proceedings of the national academy of sciences* 99(12):7821–7826.

Grover, A., and Leskovec, J. 2016. node2vec: Scalable feature learning for networks. In *Proceedings of the 22nd ACM SIGKDD international conference on Knowledge discovery and data mining*, 855–864. ACM.

He, D.; Feng, Z.; Jin, D.; Wang, X.; and Zhang, W. 2017. Joint identification of network communities and semantics via integrative modeling of network topologies and node contents. In *AAAI*, 116–124.

Huang, X.; Cheng, H.; and Yu, J. X. 2015. Dense community detection in multi-valued attributed networks. *Information Sciences* 314:77–99.

Huang, X.; Cheng, H.; and Yu, J. X. 2016. Attributed community analysis: Global and ego-centric views. *IEEE Data Eng. Bull.* 39(3):29–40.

Hunter, D. R., and Lange, K. 2004. A tutorial on mm algorithms. *The American Statistician* 58(1):30–37.

Kim, J., and Park, H. 2008. Sparse nonnegative matrix factorization for clustering. *Technical report, Georgia Institute of Technology*.

Lee, D. D., and Seung, H. S. 2001. Algorithms for non-negative matrix factorization. In *Advances in neural information processing systems*, 556–562.

Leskovec, J., and Mcauley, J. J. 2012. Learning to discover social circles in ego networks. In *Advances in neural information processing systems*, 539–547.

Mikolov, T.; Chen, K.; Corrado, G.; and Dean, J. 2013. Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Newman, M. E., and Girvan, M. 2004. Finding and evaluating community structure in networks. *Physical review E* 69(2):026113.

Perozzi, B.; Al-Rfou, R.; and Skiena, S. 2014. Deepwalk: Online learning of social representations. In *Proceedings of the 20th ACM SIGKDD international conference on Knowledge discovery and data mining*, 701–710. ACM.

Ruan, Y.; Fuhry, D.; and Parthasarathy, S. 2013. Efficient community detection in large networks using content and links. In *Proceedings of the 22nd international conference on World Wide Web*, 1089–1098. ACM.

Shi, J., and Malik, J. 2000. Normalized cuts and image segmentation. *IEEE Transactions on pattern analysis and machine intelligence* 22(8):888–905.

Wang, D.; Li, T.; Zhu, S.; and Ding, C. 2008. Multi-document summarization via sentence-level semantic analysis and symmetric matrix factorization. In *Proceedings of the 31st international ACM SIGIR conference on Research and development in information retrieval*, 307–314. ACM.

Wang, F.; Li, T.; Wang, X.; Zhu, S.; and Ding, C. 2011. Community discovery using nonnegative matrix factorization. *Data Mining and Knowledge Discovery* 22(3):493–521.

Wang, X.; Jin, D.; Cao, X.; Yang, L.; and Zhang, W. 2016. Semantic community identification in large attribute networks. *Thirtieth AAAI Conference on Artificial Intelligence*.

Wang, X.; Cui, P.; Wang, J.; Pei, J.; Zhu, W.; and Yang, S. 2017. Community preserving network embedding. In *AAAI*, 203–209.

Wu, H., and Liu, Z. 2010. Non-negative matrix factorization with constraints. In *Proceedings of the 24th AAAI Conference on Artificial Intelligence*, 506–511.

Yang, J., and Leskovec, J. 2013. Overlapping community detection at scale: a nonnegative matrix factorization approach. In *Proceedings of the sixth ACM international conference on Web search and data mining*, 587–596. ACM.

Yang, J., and Leskovec, J. 2015. Defining and evaluating network communities based on ground-truth. *Knowledge and Information Systems* 42(1):181–213.

Yang, T.; Jin, R.; Chi, Y.; and Zhu, S. 2009. Combining link and content for community detection: a discriminative approach. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining*, 927–936. ACM.

Yang, Z.; Hao, T.; Dikmen, O.; Chen, X.; and Oja, E. 2012. Clustering by nonnegative matrix factorization using graph random walk. In *Advances in Neural Information Processing Systems*, 1079–1087.

Yang, J.; McAuley, J.; and Leskovec, J. 2013. Community detection in networks with node attributes. In *2013 IEEE 13th International Conference on Data Mining*, 1151–1156.

Zhang, D.; Yin, J.; Zhu, X.; and Zhang, C. 2016. Homophily, structure, and content augmented network representation learning. In *Data Mining (ICDM), 2016 IEEE 16th International Conference on*, 609–618. IEEE.

Zhang, H.; King, I.; and Lyu, M. R. 2015. Incorporating implicit link preference into overlapping community detection. In *AAAI*, 396–402.