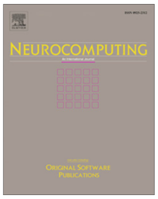
Neurocomputing 494 (2022) 336–345

Contents lists available at ScienceDirect

Neurocomputing

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



Overlapping communities detection based on cluster-ability optimization

Changjian Fang[a], Zhen-Zhou Lin[b,*]

[a] School of Information Engineering, Nanjing Audit University, Nanjing, China
[b] Experimental Teaching Center of Economics and Management, Nanjing University of Finance and Economics, Nanjing, China

ARTICLE INFO

Article history:
Received 29 December 2021
Revised 23 February 2022
Accepted 17 April 2022
Available online 20 April 2022
Communicated by Zidong Wang

Keywords:
Complex networks
Overlapping community detection
Cluster-ability
Kullback–Leibler divergence
Nonnegative matrix decomposition

ABSTRACT

Detecting overlapping communities in networks is very important to obtain and understand the overall structural characteristics of real worlds. In this paper, a new approach is proposed based on the optimization of cluster-ability to detect overlapping network communities. Specially, we design a new learning objective, i.e. the *cluster-ability*, which aims at minimizing the discrepancy between the community indicator matrix and the node similarity matrix. To deal with the optimization problems, the error is qualified by the Kullback–Leibler divergence in place of Euclidean distance. To optimize the objective function, we first use a new form of nonnegative matrix decomposition to find a solution space, and then we formulate a more appropriate and convenient multiplicative algorithm to solve the function. Finally, we systematically evaluate the proposed method on plenty of artificial networks with various network characteristics and real-world network. The results show that our method achieves the best performance on the networks with stronger overlaps, compared with the existing state-of-the-art algorithms.

1. Introduction

In recent years, due to the development of social networks and big data technology [1,2], community detection in large-scale complex networks has become a hot research area. Community detection [1–3], as a basic and important task in complex network research, aims to mine the node sets that within these sets are closely connected, while between these sets are sparsely connected. In real-world networks, some nodes may belong to multiple communities at the same time, that is, overlapping community structures. Therefore, the detection of overlapping community structure is more important and it is helpful to obtain and understand the overall structural characteristics of complex networks [4–6]. However, traditional algorithms [7–13] are no longer effective since a node may belong to multiple communities in overlapping community detection. Furthermore, the scale of complex networks is becoming huge, and the existing overlapping community mining algorithms are less efficient when dealing with such large-scale networks. Therefore, accurate and faster algorithms are needed to better cope with the problem of mining overlapping communities in large-scale complex networks.

At the same time, a number of algorithms for overlapping community detection have been developed which mainly include faction filtering method [14], edge graph partitioning method [15], local expansion method [16,17] and so on. The faction filtering algorithm (Clique Percolation Method, CPM) [14] regards the community as a set of fully connected subgroups (completed subgraphs), so it is more suitable for networks with more fully connected subgraphs. Edge graph division [15] utilizes the natural overlapping characteristics of edges, by generating the edge graphs through the transformation of node link to edge link, and then divides them to obtain the community structure. While the community expansion method based on local structural fitness (Local Fitness Maximization, LFM) [16,17] considers local structural features, and gradually expands to generate a community, and multiple expanded communities form a natural overlap. At the same time, LFM also adopts a multi-resolution strategy to assist users in making optimal choices. In addition, for different types of networks, scholars have further proposed some hybrid methods [18,19] combined with traditional non-overlapping community discovery methods [20–23]. More information can refer to revelent works and reviews [4–6,24–26]. However, most existing approaches mainly based on the traditional optimization [27–31] or heuristic methods [32–34], which do not meet both speed and accuracy requirements simultaneously.

* Corresponding author.
E-mail address: zhenz_lin@163.com (Z.-Z. Lin).

https://doi.org/10.1016/j.neucom.2022.04.091
0925-2312/

Thus, we introduce a new approach based on the optimization of cluster-ability to detect overlapping network communities. To insure the balanced community structure, we design a new learning objective, i.e. the *cluster-ability*, which aims at minimizing the discrepancy between the community indicator matrix and the node similarity matrix. To deal with the optimization problem, in our method, the error is qualified by the Kullback–Leibler divergence in place of the most common used Euclidean distance. Because we minimize the divergence over all the reasonable partitions, the most suitable number of communities is able to be chosen automatically. To optimize the objective function, an efficient algorithm is also proposed. Specifically, we first decompose the nonnegative matrix with a new form to find a solution space, and then we formulate a more appropriate and convenient multiplicative algorithm to solve the function. Finally, we systematically evaluate our approach on plenty of artificial networks with various network characteristics and real-world networks. The results show that our approach performs the best in these works on the networks with stronger overlaps, compared with the existing classical methods.

The remainder of this paper is organized as follows. We define the definition of cluster-ability in Section 2. Section 3 gives the Kullback–Leibler divergence and corresponding multiplicative algorithm to solve the function. The procedures of the algorithm and the complexity analysis are shown in Section 4. In Section 5, we verify the performance of our method on multiple networks. We finally conclude the paper Section 6.

2. The cluster-ability

2.1. Notions

For a network G with N nodes, $V_{ij} \geqslant 0$ can be defined to measure the normalized similarity between node i and j. Community detection is to divide the nodes set N into r groups (communities) with more similarity within them. Conventionally, we define the community assignments by a *nodemembershipmatrix* $M \in \{0,1\}^{N\times r}$, in which M_{ik} represents the probability that node i belongs to community C_k. Specially, as for fuzzy community detection, there is $M_{ik} \in \{0,1\}$.

On another scale, based on M, we can define the *communityindicatormatrix* as $U = MM^T, U \in \{0,1\}^{N\times N}$. Specially, only when node i and j are in the same community, there is $U_{ij} = 1$. Obviously, U is a blockwise matrix. If the community assignment is efficient, we naturally suppose the node membership matrix V is equivalent to the community indicator matrix U. There are a number of approaches to measure the discrepancy [35] between V and U, e.g. the Jaccard similarity and Cosine similarity. Among these measures, the Kullback–Leibler divergence $D(V\|U)$ is a good choice [36], which is very efficient for discrete and asymmetric data. We can utilize the Kullback–Leibler divergence $D(V\|U)$ to measure the quality of community partition.

2.2. The definition of cluster-ability

We can understand M in a probabilistic view. Here, $M_{ik} = P(k|i)$ represents the probability of the i-th node belonging to the k-th community in networks, where i, j, and v (from 1 to N) represent the indices of nodes, and k and l (from 1 to r) represent the indices of communities. With the uniform prior over nodes, i.e., $P(j) = 1/N$, we can calculate that

$$P(j|k) = \frac{P(k|j)P(j)}{\sum_{v=1}^{r} P(k|v)P(v)} = \frac{P(k|j)}{\sum_{v=1}^{r} P(k|v)} \tag{1}$$

due to the Bayes' formula. Thus,

$$\begin{aligned} U_{ij} &= \sum_{k=1}^{r} \frac{M_{ik}M_{jk}}{\sum_{v=1}^{N} M_{vk}} = \sum_{k=1}^{r} \frac{P(k|j)}{\sum_{v=1}^{N} P(k|v)} P(k|i) \\ &= \sum_{k=1}^{r} P(j|k)P(k|i) = P(j|i). \end{aligned} \tag{2}$$

Here, we define the **cluster-ability** as *the maximum proximity between* the *node similarity matrix V* and the normalized *community indicator matrix U*. Specially, we can employ $\mathscr{C}(V) \overset{\text{def}}{=} min_M D(V\|U)$ to measure the cluster-ability of a given community structure. Here, we use the adjacent matrix as node similarity matrix of a graph, which is enough to capture the feature of connections. Since the optimum is over all the possible partitions, there may have different number of communities r. Thus, it needs to learn both the community structure and the reasonable community number. To show the definition more clearly, we illustrate it in Fig. 1. Here, the node similarity matrix V represents the probability of similarity between two nodes, and the community indicator matrix U represents the probability that two nodes that share one or more communities. The cluster-ability of the community partition $\mathscr{C}(V)$ is calculated as the minimum of Kullback–Leibler divergence between the node similarity matrix V and the community indicator matrix U.

However, it is difficult to calculate the cluster-ability $\mathscr{C}(V)$ by minimize $D(V\|U)$, which is a typical NP-hard problem[37]38. Moreover, if we minimize $D(V\|U)$ directly, it will always obtain unbalanced partitions[39]. In a other word, the size of some communities will naturally much smaller than others, and these consequences are unreliable in the real world. By normalizing U that $\sum_{i=1}^{N} U_{ij} = 1$ and $\sum_{j=1}^{N} U_{ij} = 1$, or similarly by normalizing M with $M_{ik} = M_{ik}/\sqrt{\sum_{v=1}^{N} M_{vk}}$, we can get more balanced partitions.

3. Matrix decomposition and cluster-ability optimization

In this section, we firstly present the process of minimizing $D(V\|U)$ (optimize the cluster-ability) over by choosing appropriate divergence measure. Then we propose a multiplicative minimization algorithm to search the stationary point of the objective function.

3.1. Kullback–Leibler divergence

The conventional way that measuring the discrepancy is using Euclidean distance, or Frobenius norm. However, in many real applications, the features of raw data are always weakly informative, therefore it is not suitable to use the Euclidean distance in these situations. We note that if we want to calculate the similarity using some simple measures such as Hamming distance or Euclidean distance, it is precise only when the neighborhood is very small. However, the nodes that belong to different communities often become mixing in very dense networks.

Therefore, we use a more proper index, i.e. Kullback–Leibler (KL) divergence, to measure the approximation error, since this approximation relies heavily on some large values of V [36]. To model the rare occurrences of reliable similarities more appropriately, we utilize the underlying poisson likelihood, and employ the following an optimization problem to formulate the objective function:

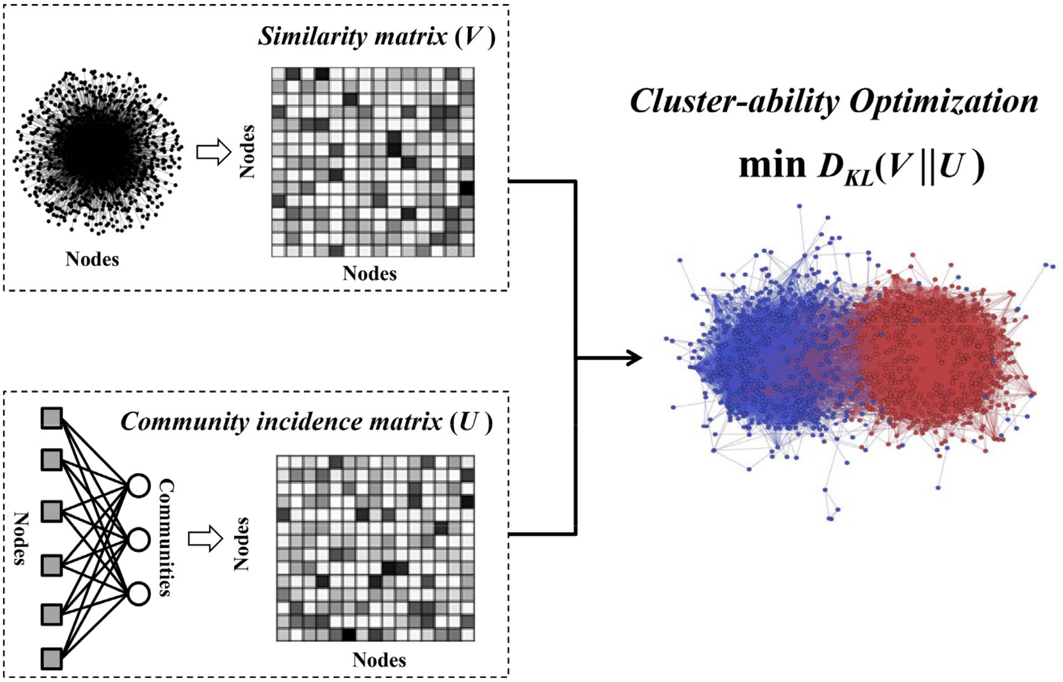


Fig. 1. The illustration of detecting overlapping communities based on cluster-ability optimization. In this figure, the matrix U represents the probability that two nodes share one or more communities; while the matrix U represents the similarity of two nodes. The cluster-ability denotes the maximum proximity between the node similarity matrix V and the normalized community indicator matrix U. The the communities can be uncovered by evaluating the Kullback–Leibler divergence $\mathscr{C}(V|r) = \min D_{KL}(V\|U)$.

$$\begin{aligned} \min_{M \geqslant 0} D_{\mathrm{KL}}(V\|U) \quad &= \sum_{i=1}^{N}\sum_{j=1}^{N}\left(V_{ij}\log\tfrac{V_{ij}}{U_{ij}} - V_{ij} + U_{ij}\right) \\ \text{subjectto} \qquad U_{ij} &= \sum_{k=1}^{r}\frac{M_{ik}M_{jk}}{\sum_{v} M_{vk}}, \\ &\sum_{k=1}^{r} M_{jk} = 1, i = 1, \ldots, N. \end{aligned} \tag{3}$$

Here, we neglect the constant terms in $D_{\mathrm{KL}}(V\|U)$ due to $\sum_{i=1}^{N}\sum_{j=1}^{N}U_{ij} = N$, and the objective function is equivalent to maximize $\sum_{i=1}^{N}\sum_{j=1}^{N}V_{ij}\log U_{ij}$. Beside the decomposition form in U, Eq. (3) has very similar form with PLSI [40].

We can find that the Kullback Leibler (KL) divergence is also favorable to the optimization, which is shown below in the algorithm development. It is explicit that only involving the non-zero elements in the similarity matrix V will make the implementation more efficiently in objective and gradient calculation [41]. Furthermore, by breaking it into some additive terms, the complex structure of U is resolved, and the objective function is optimize by using the convex-concave procedure [42].

3.2. Regularization

In the objective function (3), it uses the matrix M as the parameter, where its rows sum equals to one. Supposing that the rows can be obtained by observing a common Dirichlet distribution, then the complexity of M can be controlled using the log-Dirichlet prior. Thus the cost function has the following form:

$$\mathscr{F}(M) = -\sum_{i=1}^{N}\sum_{j=1}^{N}V_{ij}\log U_{ij} - (\alpha - 1)\sum_{i=1}^{N}\sum_{k=1}^{r}\log M_{ik}. \tag{4}$$

when the value of V_{ij} are integers. Specially, the parameter α controls the degree of regularization of cost function (4). When $\alpha = 1$, the Dirichlet prior vanishes; while when $\alpha > 1$, the prior gives wider relaxation by smoothing the entries of matrix M, which is normally desired in the early phases of M learning. Similarly, we can also use a total Shannon information term to make regularization.

3.3. Optimization

To solve the optimization for nonnegative matrix factorization (NMF) problems [43], we use the widely used multiplicative update rules. To minimize the objective function $\mathscr{F}$ which parameterized by the nonnegative matrix M, the gradient is calculated firstly and divided into two nonnegative parts ($\nabla_{ik}^{+} \geqslant 0$ and $\nabla_{ik}^{-} \geqslant 0$):

$$\nabla_{ik} \overset{\text{def}}{=} \frac{\partial \mathscr{F}}{\partial M_{ik}} = \nabla_{ik}^{+} - \nabla_{ik}^{-}. \tag{5}$$

Generally speaking, we can identify these two parts easily from the gradient. Next we apply the algorithm repeatedly by a multiplicative update rule, i.e., $M_{ik} \leftarrow M_{ik}\frac{\nabla_{ik}^{-}}{\nabla_{ik}^{+}}$, until to the convergence. Such algorithms have some excellent properties, i.e., they keep M to be positive and don't need to tune the size of learning step. For all kinds of NMF problems, after each iteration, $\mathscr{F}$ decreases monotonically by the multiplicative update rules, and M thus converges to a fixed point.

For the above multiplicative update rules, we cannot apply it straightforwardly due to the probability constrains on the M rows. In practice, it may obtain poor partition results if we project the rows of M to the probability simplex. Thus, to deal with the probability constraint more efficiently, we apply a new relaxing strategy. Firstly, we put forward Lagrangian multipliers $\{\lambda_i\}_{i=1}^{N}$ of the constraint:

$$\mathscr{H}(M,\lambda) = \mathscr{F}(M) + \sum_{i}\lambda_i\left(\sum_{k=1}^{r}M_{ik} - 1\right). \tag{6}$$

For M, Eq. (6) indicates an early multiplicative update rule:

$$M'_{ik} = M_{ik} \frac{\nabla^{-}_{ik} - \lambda_i}{\nabla^{+}_{ik}}, \tag{7}$$

where

$$\frac{\partial \mathscr{F}}{\partial M} = \underbrace{\left[\left(M^T Z M\right)_{kk} s_k^{-2} + M_{ik}^{-1}\right]}_{\nabla_k^{+}} - \underbrace{\left[2(ZM)_{ik} s_k^{-1} + \alpha M_{ik}^{-1}\right]}_{\nabla_k^{-}}, \tag{8}$$

with $Z_{ij} = V_{ij}/U_{ij}$ and $s_k = \sum_{v=1}^{N} M_{vk}$. Imposing $\sum_k M'_{ik} = 1$ and isolating λ_i, there is

$$\lambda_i = \frac{b_i - 1}{a_i}, \tag{9}$$

where

$$a_i = \sum_{l=1}^{r} \frac{M_{il}}{\nabla^{+}_{il}}, \text{and} b_i = \sum_{l=1}^{r} M_{il} \frac{\nabla^{-}_{il}}{\nabla^{+}_{il}}. \tag{10}$$

Putting this λ into Eq. (7), there is

$$M_{ik} \leftarrow M_{ik} \frac{\nabla^{-}_{ik} a_i + 1 - b_i}{\nabla^{+}_{ik} a_i}. \tag{11}$$

We add b_i to both the denominator and numerator to keep M positive, thus we obtain the final update rule with the same fixed point:

$$M_{ik} \leftarrow M_{ik} \frac{\nabla^{-}_{ik} a_i + 1}{\nabla^{+}_{ik} a_i + b_i}. \tag{12}$$

We summarize the above steps in Algorithm 1 in the next section. There is no necessity to formulate the whole matrix U in practise, and the ratio $Z_{ij} = V_{ij}/U_{ij}$ just requires to calculate the non-zero V entries.

Remark 1. Denoting that M^{new} is the updated matrix after each iteration in Algorithm 1. There is $\mathscr{H}(M^{new}, \lambda) \leqslant \mathscr{H}(M, \lambda)$ with $\lambda_i = (b_i - 1)/a_i$, which guarantee of monotonicity which the algorithm obeys. Here, b_i represents the sum of the rows of unconstrained multiplicative results, while a_i represents the balance between the probability simplex attraction and the gradient learning force. In addition to convenience in computation, it can be found that this relaxation strategy is more robust than brute-force projection after each step.

4. The algorithm

In order to actually execute the community detection in networks, an iterative algorithm (Algorithm 1) is proposed below, in which the detailed procedures are described step-by-step.

```
Algorithm 1: Overlapping community detection based on
cluster-ability optimization

Input: The similarity matrix $V$, the number of communities $r$,
  the initial of $M$.
Output: The community membership matrix $X$;
Repeat
1: $U_{ij} = \sum_{k=1}^{r} \frac{M_{ik} M_{jk}}{\sum_{v=1}^{N} M_{vk}}$;
2: $Z_{ij} = V_{ij}/U_{ij}$;
3: $s_k = \sum_{v=1}^{N} M_{vk}$
4: $\nabla_k^{+} = \left(M^T Z M\right)_{kk} s_k^{-2} + M_{ik}^{-1}$;
5: $\nabla_k^{-} = 2(ZM)_{ik} s_k^{-1} + \alpha M_{ik}^{-1}$;
6: $a_i = \sum_{l=1}^{r} \frac{M_{il}}{\nabla^{+}_{il}}$, and $b_i = \sum_{l=1}^{r} M_{il} \frac{\nabla^{-}_{il}}{\nabla^{+}_{il}}$;
```

a (*continued*)

```
Algorithm 1: Overlapping community detection based on
cluster-ability optimization

7: $M_{ik} \leftarrow M_{ik} \frac{\nabla^{-}_{ik} a_i + 1}{\nabla^{+}_{ik} a_i + b_i}$.
until $M$ converges under the given tolerance.
```

Before applying the algorithm, one important question that should be addressed is determining the number of communities in complex networks. However, the number of communities in networks is not known as a priori information in advance, but must be chosen from the number of ground truth classes in a certain range. Here, we run Algorithm 1 with variable r values, and then after discretizing M to the matrix of community indicator, compute the corresponding residual $D(V\|M)$. Finally, the optimal number of communities r with the lowest residual $D(V\|M)$ can be chosen.

For any community detection method of non-convex optimization, it is significant to have a favorable initialization. Our algorithm can set any partition results as its start. In this paper, a little positive disturbance (e.g. 0.2) is added to all entries of the initial community indicator matrix. After that, we feed the perturbed matrix to the optimization(with α = 1 in cost function (3)). We select the optimal partition result when the $D(V\|M)$ is smallest among all runs of Algorithm 1.

Specially, the parameter α controls the degree of regularization of cost function (3). When $\alpha = 1$, the Dirichlet prior disappears; while when $\alpha > 1$, the prior gives wider relaxation by smoothing the entries of matrix M, which is normally desired in early phases of M learning. Particularly, Non-regularized initialization (i.e. with α = 1) can also be achieved using the regularized parameter (i.e. with various $\alpha \neq 1$).

For the time complexity of Algorithm 1, it is mainly spent on updating $U_{ij}, \nabla_k^{+}, \nabla_k^{-}$ and M_{ik}. When updating U_{ij} takes the time complexity in $O\left(rN^2\right)$, where r denotes the number of communities and N represents the number of nodes(the dimension of matrix M); when updating ∇_k^{+} and ∇_k^{-}, since we need to compute the inverse matrix of M, the time complexity is $O\left(N^2 \log N\right)$. For M_{ik}, updating it takes N^2 time. In conclusion, the time complexity of one iteration in Algorithm 1 is $O\left(N^2 \log N\right)$.

5. Experimental results

We put the proposed algorithm to the test on both artificial and real-world networks, and compare it with the state of art algorithms for overlapping community detection. All these operations are performed on a PC with a 4 GHz CPU, 2 GB of memory, Windows 10 operating system and Matlab 2018b.

5.1. Criteria for evaluation

In this paper, two evaluation indexes are used, which are normalized mutual information *NMI* [44] and improved overlapping modularity Q_{ov} [45]. For datasets with explicit community structure, NMI is used as the evaluation criterion, otherwise Q_{ov} is used as the evaluation criterion. The standardized mutual information *NMI* is a measure of the experimental results' resemblance to the explicit community structure. It is calculated as follows.

$$NMI = \frac{-2\sum_{i=1}^{CN_R}\sum_{j=1}^{CN_A} B_{ij}\log_2\left(\frac{B_{ij}*N}{B_{i.}*B_{.j}}\right)}{\sum_{i=1}^{CN_R} B_{i.}\log_2\left(\frac{B_{i.}}{N}\right) + \sum_{j=1}^{CN_A} B_{.j}\log_2\left(\frac{B_{.j}}{N}\right)}. \tag{13}$$

In the formula, A represents the community structure divided by the algorithm, and R represents the real community structure; CN_A indicates the number of communities in the algorithm-generated community structure; CN_R represents the number of real divided communities; B represents the mixing matrix; B_{ij} represents the number of nodes existing in both the i-th and j-th community; $B_{i.}(B_{.j})$ denotes the sum of the elements in the matrix's i-th row (j-th column); and N denotes the total number of nodes in the network.

For datasets without the known community structure, the quality of the algorithm is measured using the improved overlap modularity degree Q_{ov}. The improved overlap modularity Q_{ov} is calculated as follows:

$$Q_{ov} = \frac{1}{2m}\sum_{c\in C}\sum_{i,j\in V}\left[q_{ijc}A_{ij} - p_{ijc}\frac{k_i k_j}{2m}\right]. \tag{14}$$

Here, $q_{ijc} = \Psi(\omega_{i,c}, \omega_{j,c})$ denotes the probability that both node i and node j belong to community c, $\omega_{i,c}$ denotes the probability that node i belongs to community c, and p_{ijc} denotes the probability that either node i or node j belongs to community c which is calculated as follows:

$$p_{ijc} = \frac{\sum_{j\in V}\Psi(\omega_{i,c}, \omega_{j,c})}{|V|} * \frac{\sum_{i\in V}\Psi(\omega_{i,c}, \omega_{j,c})}{|V|}. \tag{15}$$

Here, $\Psi(\omega_{i,c}, \omega_{j,c})$ is calculated as follows:

$$\Psi(\omega_{i,c}, \omega_{j,c}) = \frac{1}{\left(1+e^{-f(\omega_{i,c})}\right)\left(1+e^{-f(\omega_{j,c})}\right)}. \tag{16}$$

The function in Eq. (16) is defined as: $f(x) = 60x - 30$.

Table 1
Parameters and meaning of LFR benchmark network.

Parameters	Meaning
N	Number of Nodes
d	Average Degree of Network
$d_{\max}$	Maximum Degree of Network
γ	Distribution Parameter of Nodes' Degree
β	Distribution Parameter of Community Scale
$\min_c$	Number of Nodes in the Minimum Community
$\max_c$	Number of Nodes in the Maximum Community
On	Number of Overlapping Nodes
Om	Number of Overlapping Communities
μ	Mixing Parameter

5.2. Artificial network dataset

To verify the performance of our method, we use the LFR artificial network [50] which organized with the scale-free distribution of vertex degree and community scale. Therefore, compared with other standard and uniform benchmarks, it is a more rigorous verification for the overlapping community detection. It has a few key parameters to control the topological properties: the number of nodes N, the average vertex degree d, the maximal vertex degree $d_{\max}$, two parameters γ and β which control the distribution of community structure, the minimal size of community $\min_c$ and the maximal size of community $\max_c$, the number of overlapping nodes On and the mixing parameter μ. The mixing parameter μ changes in the range of [0,1], which determines the vagueness of the network community. The detailed parameters and their meanings are shown in Table 1.

In this section, we change the parameters of LFR artificial network and generate eight sets of overlapping artificial networks which denoted as G1 to G8. Each set of network data contains eight networks with parameters O_m taking values from 2 to 9 respectively, and the rest of the parameters are the same. In Table 2, we show the parameter settings of each group of overlapping artificial networks. In the experiments, the proposed algorithm will be compared with overlapping community detection algorithms such as LFM [17], GCE [46], OCDDP [47], CMNN [48], COPRA [20], CFinder [14], and LINK [49], using *NMI* as the evaluation criterion.

Fig. 2 shows the performance of each algorithm on the overlapping artificial networks G1 to G8, where the vertical coordinate is the normalized mutual information *NMI* value and the x coordinate indicates the number of overlapping communities O_m. On all overlapping artificial networks, our algorithm outperforms the other comparison algorithms, as shown in Fig. 2. Compared with the LFM algorithm, the difference in *NMI* between the two is increasing as O_m increases. If the mixing parameter μ as well as O_m is kept constant and the number of overlapping nodes O_n is used as a variable, for example, G1 and G3 networks and fixing O_m, it can be seen that the *NMI* difference between our method and LFM algorithm is increasing as O_n increases. When the mixing degree μ increases, the *NMI* difference between our method and LFM algorithm is increasing. For example, comparing the performance of both on the G1 and G2 networks. From the above analysis, it is clear that our method is more suitable for community discovery on networks with higher overlap and mixing parameter μ compared to LFM algorithm.

The comparison of proposed method and other algorithms have the similar results. The OCDDP [24] and CMNN [25] algorithms are both based on the idea of density peaks, and both achieve good performance in networks with low mixing parameter μ, which is not much different from the our algorithm. While when the mixing parameter μ increases, the performances of both the CMNN and OCDDP algorithms are significantly worse. In similar, the CFinder [14] and COPRA [26] algorithms are highly sensitive to the mixing parameter μ, but the magnitude is not as large as the CMNN and

Table 2
Parameter settings for each group of overlapping artificial networks.

ID	*N*	*d*	$d_{\max}$	γ	β	$\min_c$	$\max_c$	*On*	μ
G1	5000	10	50	2	1	10	100	500	0.1
G2	5000	10	50	2	1	10	100	500	0.3
G3	5000	10	50	2	1	10	100	1000	0.1
G4	5000	10	50	2	1	10	100	1000	0.3
G5	5000	10	50	2	1	20	200	500	0.1
G6	5000	10	50	2	1	20	200	500	0.3
G7	5000	10	50	2	1	20	200	1000	0.1
G8	5000	10	50	2	1	20	200	1000	0.3

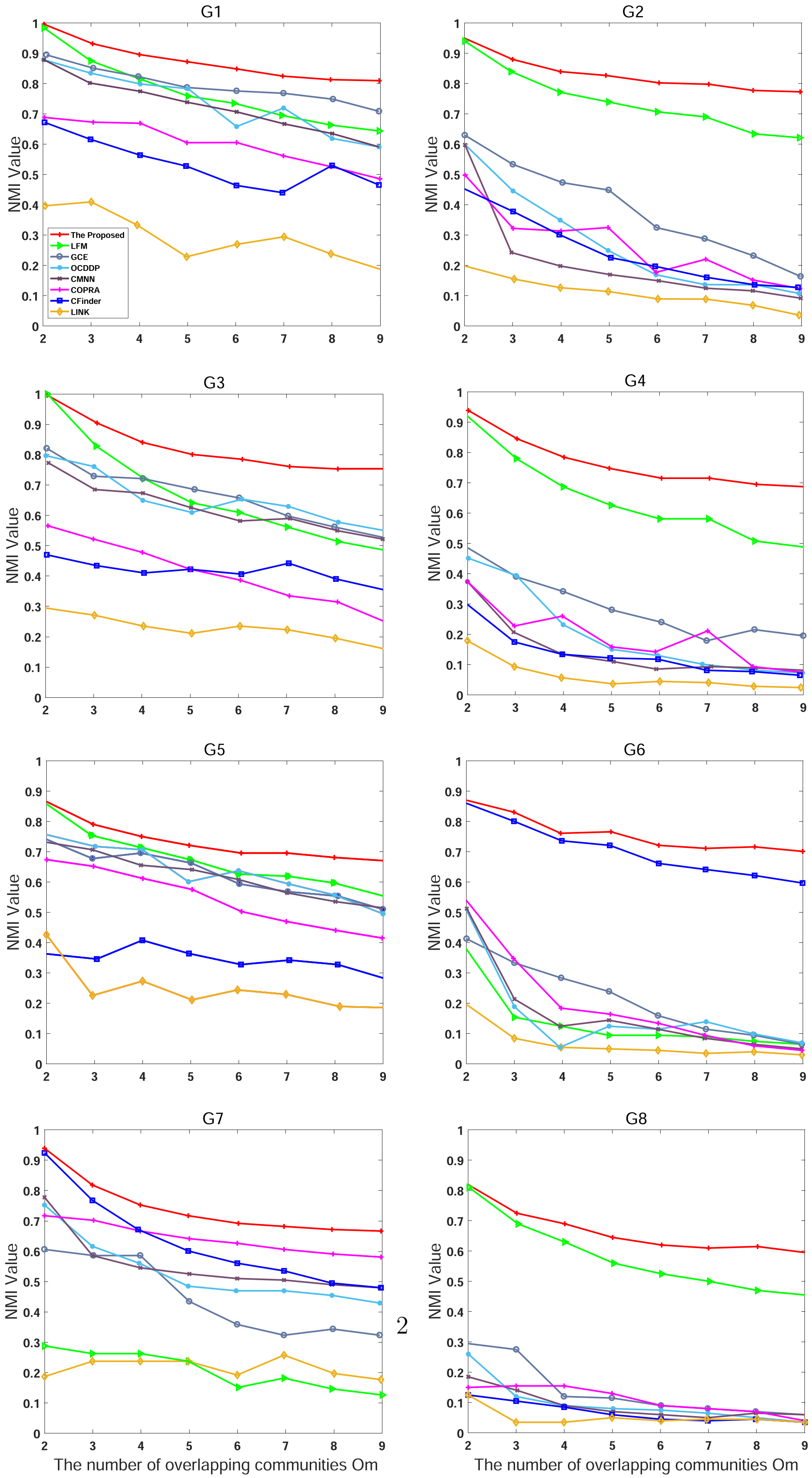


Fig. 2. Performance of each algorithm in G1-G8 networks. Here, we compared with 8 overlapping community detection algorithms including LFM, GCE, OCDDP, CMNN, COPRA, CFinder, and LINK, using *NMI* as the evaluation criterion.

Table 3
The topological properties of 9 real networks.

Network	Number of nodes	Number of links
Karate	34	78
Dolphin	62	159
Pol. Books	105	441
Football	115	616
Jazz	198	2742
Email	1133	5451
Pol. Blogs	3982	6803
Power	4941	6593
PGP	10680	24316

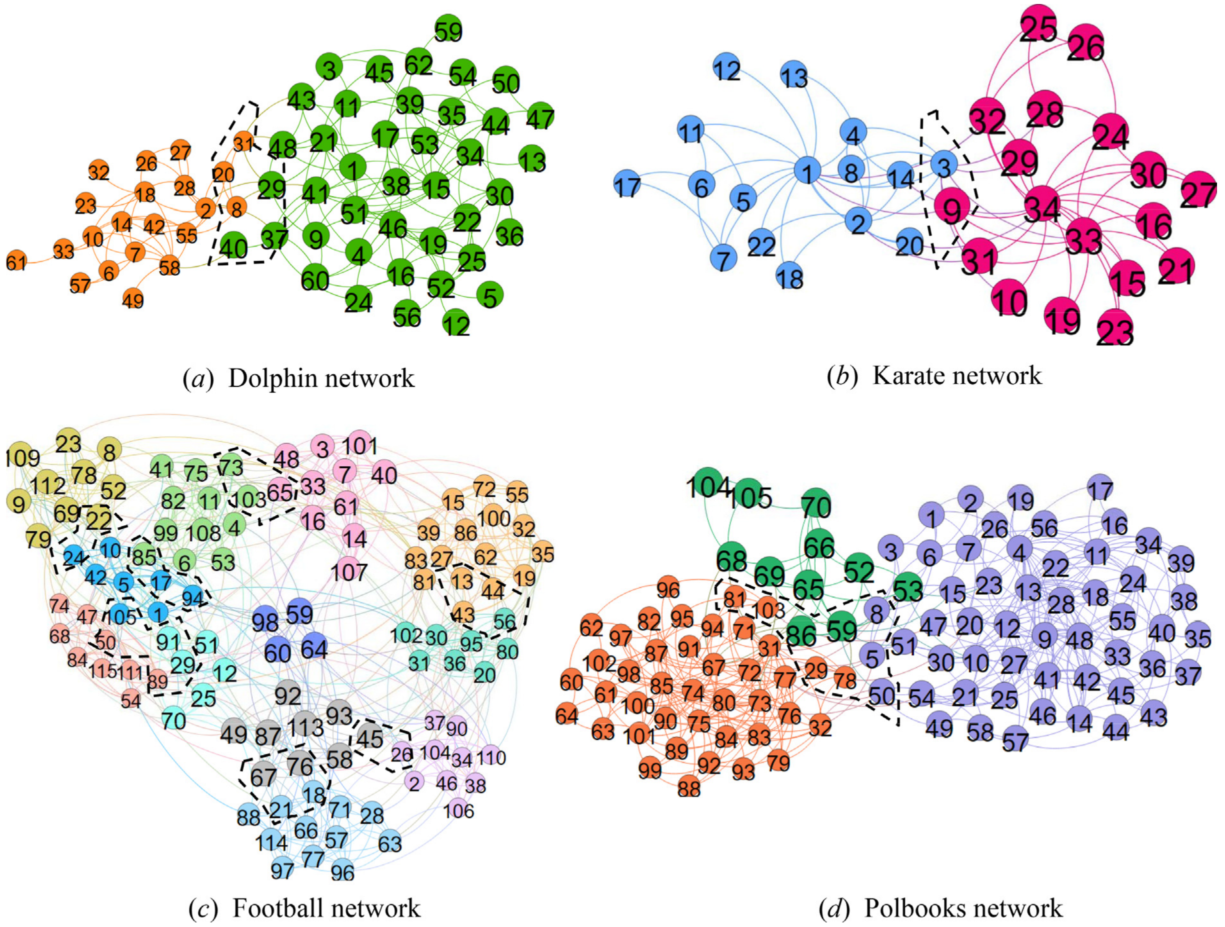
OCDDP algorithms. The GCE [46] and LINK [49] algorithms perform relatively stable in networks with different mixing parameter μ, on the condition that the GCE algorithm has good performance in all networks while the LINK algorithm has the worst performance among all algorithms.

5.3. Real network dataset

The experiments in this section are divided into two parts, which are small-scale real networks and large-scale real networks experiments.

Table 4
The comparison of our method's experimental outcomes to those of other algorithms using Q_{ov}.

Network	The Proposed	GCE	OCDDP	SLPA	COPRA	SpeakEasy	Multiscale	CMNN
Karate	**0.740**	0.576	0.704	0.698	0.379	0.702	0.248	/
Dolphin	**0.776**	0.662	0.774	0.760	0.662	0.697	0.377	/
Football	0.724	0.688	0.720	0.699	0.692	0.691	0.459	**0.725**
Pol. Books	**0.847**	0.833	0.840	0.830	0.828	0.684	0.135	0.846
Jazz	0.722	0.722	0.626	0.704	0.715	0.640	0.104	**0.734**
Email	**0.642**	0.529	**0.642**	0.631	0.519	0.511	0.069	0.626
Power	**0.915**	0.069	0.908	0.659	0.476	0.675	0.542	0.897
Pol. Blogs	0.799	0.800	0.800	0.800	0.800	0.766	/	**0.814**
PGP	**0.785**	0.664	0.732	/	0.775	/	/	0.763



(a) Dolphin network

(b) Karate network

(c) Football network

(d) Polbooks network

Fig. 3. The visualization of experimental results of our method on 4 real-world networks. Here, these four networks include (a) Dolphin network, (b) Karate network, (c) Football network and (d) Polbook network. The overlapping parts are highlighted by the dashed lines.

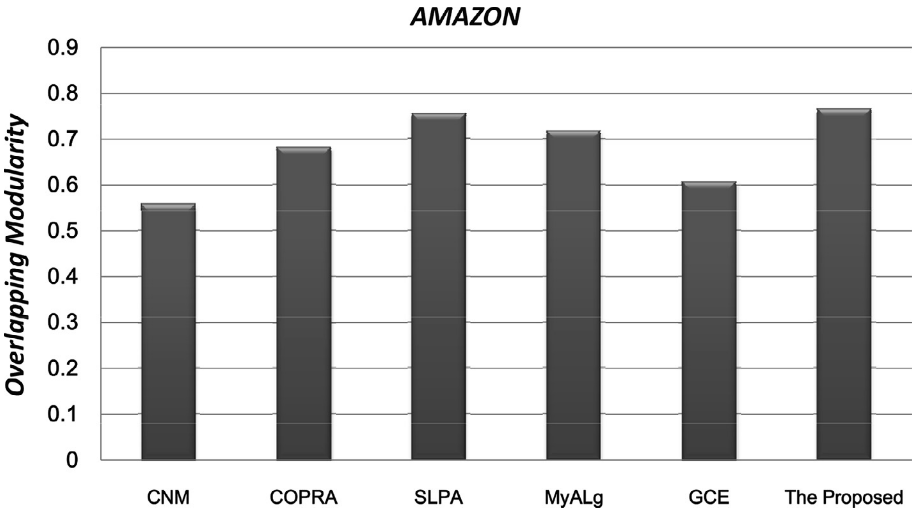


Fig. 4. The comparison of experimental results in AMAZON network between our method and other algorithms using Q_{ov}.

Table 5
The performance evaluation of community detection based on Variation of Information (VOI).

Datasets	AMAZON	Livejournal	Polblog
CNM	0.723	0.609	0.573
COPRA	0.227	0.461	0.223
SLPA	0.319	0.678	0.257
GCE	0.563	0.656	0.321
MyAlg	0.313	0.271	0.289
The Proposed	0.118	0.101	0.219

5.3.1. Small-scale real networks

In order to compare with algorithms such as GCE [46], OCDDP [24], SLPA [21], COPRA [20], SpeakEasy [51], Multiscale [52], and CMNN [48], nine real network datasets are used in this section, including Karate, Dolphin, Football, Pol. Books, Jazz, Email Power, Pol. Blogs and PGP. Considering that networks such as Karate, Dolphin, Football, and Pol. Books as having some overlapping structures, the experiments in this section also use these four datasets for comparison experiments. Because the true overlapping community structure of these networks is unknown, the improved overlap modularity Q_{ov} is used as an evaluation criterion in this section of the experiments. Table 3 summarizes the attributes of these nine real networks.

Table 4 shows the comparison of experimental results between these algorithms. The optimal values in Table 4 have been bolded, where / stands for the original literature does not give the experimental results. It can be seen from Table 4 that our algorithm achieves the optimal Q_{ov} on six real networks, which are Karate, Dolphin, Pol. Books, Email, Power, and PGP networks. In the Football network, the Q_{ov} of our algorithm is 0.724, which is only 0.001 lower than the best performing CMNN algorithm. In the Jazz network, the Q_{ov} of our algorithm is 0.722, which is only 0.012 lower than the best performing CMNN algorithm. In the Pol.Bolgs network, our algorithm obtains a Q_{ov} of 0.799, which is only 0.015 lower than the best-performing CMNN algorithm, so it can be assumed that they obtain an approximate high-quality overlapping community structure in all three datasets. In conclusion, the proposed algorithm has a superior performance over other methods on almost all scenarios.

For the visualization of the community structure, Fig. 3(a)-(d) show the communities identified by our method on the four networks respectively. The overlapping parts between communities are highlighted by the black dash links, and the identified overlapping communities are separated by different colors. For example, in the Dolphin network (see Fig. 3(a)), node 8, 20, 29, 31, 37, 40 are shared by the two groups of dolphins. In the Karate network (see Fig. 3(b)), we observe that nodes 3 and 9 are shared by two communities identified by our method.

5.3.2. Large-scale real networks

For comparison with the CNM [53], COPRA [20], SLPA [21], GCE [46] and MyAlg [54] algorithms are applied on the large scale AMAZON network dataset. This network contains 334863 nodes and 925872 links, with 667129 triangles. Its average clustering coefficient is 0.3967 and diameter (longest shortest path) is 44. Here, we use the improved overlap modularity Q_{ov} as the evaluation criterion. Fig. 4 illustrates the experimental results of these algorithms in AMAZON network. From Fig. 4, one can observe all three algorithms obtained $Q_{ov} > 0.7$. We find that our algorithm performs the best on the AMAZON network, followed by the SLPA and MyAlg algorithms.

Finally, Variation of Information (VOI) is utilized to measure of how much information is not overlapped between different groups of community and future quantify the robustness of graphs. For VOI measure, the value 0 denotes the same community, and the value 1 denotes the most different communities. In order to evaluate the robustness of community distribution, the network links are randomized with the probability of disturbing, and the VOI between the original network and the disturbed network is calculated on a certain range of disturbing probability. We compare the different algorithms on three large scale networks and the experimental results are shown in Table 5. We can find the VOI values of the proposed algorithm are the lowest, which verify the robustness of our method.

6. Conclusion

In this paper, we proposed a new community detection algorithm based on the optimization of *cluster-ability*, which aims at minimizing the discrepancy between the matrix of community indicator and the matrix of node similarity. To optimize the Kullback–Leibler divergence, we first use a new form of nonnegative matrix decomposition to find a solution space, and then we formulate a more appropriate and convenient multiplicative algorithm to solve the function. We test our method on a large number of artificial networks with a variety of network properties, as well as real-world networks. When compared to existing state-of-the-art algorithms, the results show that our method outperforms them on networks with stronger overlaps.

In the future work, we will focus on several challenges such as extremely large scale networks including more than millions node. Furthermore, some high-order types networks such as signed networks and hyperbolic network which consist of different type links are also should be considered. In addition, the overlapping structure plays a key role in network evaluation and community evolution, which will lead to the fusion or split of the community on time, which will be studied in our future works.

CRediT authorship contribution statement

Changjian Fang: Methodology, Visualization, Software, Formal analysis, Writing - original draft. **Zhen-Zhou Lin:** Investigation, Methodology, Visualization, Software, Writing - original draft, Funding acquisition.

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.

Acknowledgement

We are grateful to the anonymous reviewers for their valuable suggestions which are very helpful for improving the manuscript. This paper is supported by the National Natural Science Foundation of China (NSFC) under Grant 72072091 and Industry Projects in Jiangsu S&T Pillar Program under Grant No. BE2019110.

References

[1] S.H. Strogat, Exploring complex networks, Nature 410 (2001) 268–276.

[2] R. Albert, A.-L. Barabasi, Statistical mechanics of complex networks, Rev. Mod. Phys. 74 (2002), 47–47.

[3] M. Girvan, M.E.J. Newman, Community structure in social and biological networks, Proc. Natl. Acad. Sci. 99 (2002) 7821–7826.

[4] S. Fortunato, Community detection in graphs, Phys. Rep. 486 (2010), 75-74.

[5] S. Fortunato, D. Hric, Community detection in networks: A user guide, Phys. Rep. 659 (2016) 1–4.

[6] Y. Fang, X. Huang, L. Qin, Y. Zhang, W. Zhang, R. Cheng, X. Lin, A survey of community search over big graphs, VLDB J. 29 (8) (2020) 353–392.

[7] H.J. Li, L. Wang, Y. Zhang, M. Perc, Optimization of identifiability for efficient community detection, New J. Phys. 22 (6) (2020) 063035.

[8] J. Cao, D. Ding, J. Liu, E. Tian, S. Hu, X. Xie, Hybrid-triggered-based security controller design for networked control system under multiple cyber attacks, Inf. Sci. 548 (2021) 69–84.

[9] J. Cao, Z. Bu, Y. Wang, H. Yang, J. Jiang, H.J. Li, Detecting prosumer-community groups in smart grids from the multiagent perspective, IEEE Trans. Systems, Man, Cybernetics: Systems 49 (8) (2019) 1652–1664.

[10] J. Cao, Y. Wang, J. He, W. Liang, H. Tao, G. Zhu, Predicting grain losses and waste rate along the entire chain: a multitask multigated recurrent unit autoencoder based method, IEEE Trans. Industr. Inf. 17 (6) (2020) 4390–4400.

[11] Z. Li, S. Zhang, R.-S. Wang, X.-S. Zhang, L. Chen, Quantitative function for community detection, Phys. Rev. E 77 (2008) 036109.

[12] M. Rosvall, C.T. Bergstrom, An information-theoretic framework for resolving community structure in complex networks, Proc. Natl. Acad. Sci. 104 (2007) 7327–7331.

[13] M. Rosvall, C.T. Bergstrom, Maps of information flow reveal community structure in complex networks, Proc. Natl. Acad. Sci. 105 (2008) 1118–1123.

[14] G. Pallal et al., Uncovering the overlapping community structure of complex networks in nature and society, Nature 435 (7043) (2005) 814–818.

[15] Y.-Y. Ahn, J.P. Bagrow, S. Lehmann, Link communities reveal multiscale complexity in networks, Nature 466 (7307) (2010) 761–764.

[16] A. Lancichinetti, F. Radicchi, J.J. Ramasco, S. Fortunato, Finding statistically significant communities in networks, PLoS ONE 6 (4) (2011) e18961.

[17] A. Lancichinetti, S. Fortunato, J. Kertesz, Detecting the overlapping and hierarchical community structure in complex networks, New J. Phys. 11 (2009) 033015.

[18] W. Chen, Z. Liu, X. Sun, Y. Wang, A game-theoretic framework to identify overlapping communities in social networks, Data Min. Knowl. Disc. 21 (2) (2010) 224–240.

[19] I. Psorakis, S. Roberts, M. Ebden, B. Sheldon, Overlapping community detection using Bayesian non-negative matrix factorization, Phys. Rev. E 83 (2011) 066114.

[20] S. Gregory, Finding overlapping communities in networks by label propagation, New J. Phys. 12 (2010) 103018.

[21] J. Xie, B.K. Szymanski, X. Liu, SLPA: Uncovering Overlapping Communities in Social Networks via a Speaker-Listener Interaction Dynamic Process, 2011 IEEE 11th International Conference on Data Mining Workshops, 11-11 Dec. 2011, Vancouver, BC, Canada.

[22] J. Xie, B.K. Szymanski, Community detection using a neighborhood strength driven Label Propagation Algorithm, 2011 IEEE Network Science Workshop, 22-24 June 2011, West Point, NY, USA.

[23] J. Xie, B.K. Szymanski, Towards linear time overlapping community detection in social networks, PAKDD 2012: Advances in Knowledge Discovery and Data Mining, pp. 25–36, Kuala Lumpur, Malaysia, May 29 C June 1, 2012.

[24] X. Zeng, W. Wang, C. Chen, G.G. Yen, A consensus communitybased particle swarm optimization for dynamic community detection, IEEE Trans. Cybern. 50 (6) (2020) 2502–2513.

[25] X. Zhang, K. Zhou, H. Pan, L. Zhang, X. Zeng, Y. Jin, A network reduction-based multiobjective evolutionary algorithm for community detection in large-scale complex networks, IEEE Trans. Cybern. 50 (2) (2020) 703–716.

[26] X. Teng, J. Liu, M. Li, Overlapping community detection in directed and undirected attributed networks using a multiobjective evolutionary algorithm, IEEE Trans. Cybern. 51 (1) (2021) 138–150.

[27] W. Yue, Z. Wang, J. Zhang, X. Liu, X, An overview of recommendation techniques and their applications in healthcare, IEEE/CAA J. Automatica Sinica 8 (4) (2021) 701–717.

[28] N. Zeng, Z. Wang, W. Liu, H. Zhang, K. Hone, X. Liu, A Dynamic Neighborhood-Based Switching Particle Swarm Optimization Algorithm, IEEE Trans. Cybern. (2020), https://doi.org/10.1109/TCYB.2020.3029748.

[29] X. Luo, Y. Yuan, S. Chen, N. Zeng, Z. Wang, Position-Transitional Particle Swarm Optimization-incorporated Latent Factor Analysis, IEEE Trans. Knowl. Data Eng. (2020), https://doi.org/10.1109/TKDE.2020.3033324.

[30] P. Wu, H. Li, N. Zeng, F. Li, FMD-Yolo: An efficient face mask detection method for COVID-19 prevention and control in public, Image Vis. Comput. 117 (2022) 104341.

[31] N. Zeng, D. Song, H. Li, Y. You, Y. Liu, F.E. Alsaadi, A competitive mechanism integrated multi-objective whale optimization algorithm with differential evolution, Neurocomputing 432 (2021) 170–182.

[32] W. Liu, Z. Wang, Y. Yuan, N. Zeng, K. Hone, X. Liu, A Novel Sigmoid-Function-Based Adaptive Weighted Particle Swarm Optimizer, IEEE Trans. Cybern. 51 (2) (2021) 1085–1093.

[33] W. Liu, Z. Wang, Y. Yuan, F.E. Alsaadi, X. Liu, A novel randomised particle swarm optimizer, Int. J. Mach. Learn. Cybern. 12 (2) (2021) 529–540.

[34] W. Liu, Z. Wang, N. Zeng, A.F.E. Alsaadi, X. Liu, A PSO-based deep learning approach to classifying patients from emergency departments, Int. J. Mach. Learn. Cybern. 12 (7) (2021) 1939–1948.

[35] F. Pompili, N. Gillis, P. Absil, and F. Glineur. ONP-MF: An orthogonal nonnegative matrix factorization algorithm with application to clustering, In European Symposium on Artificial Neural Networks, pages 297–302, 2013.

[36] R. Zass and A. Shashua. Doubly Stochastic Normalization for Spectral Clustering, Advances in Neural Information Processing Systems (NIPS), 2006.

[37] D. Aloise, A. Deshpande, P. Hansen, P. Popat, NP-hardness of euclidean sum-of-squares clustering, Machine Learning 75 (2) (2009) 245–248.

[38] M. Mahajan, P. Nimbhorkar, and K. Varadarajan. The planar k-means problem is np-hard. In Lecture Notes in Computer Science, 5431, 274–285. Springer, 2009.

[39] J. Shi, J. Malik, Normalized cuts and image segmentation, IEEE Trans. Pattern Anal. Mach. Intell. 22 (8) (2000) 888–905.

[40] T. Hofmann, Probabilistic latent semantic indexing, in: In International Conference on Research and Development in Information Retrieval (SIGIR), 1999, pp. 50–57.

[41] C. Ding, X. He, H. Simon, On the equivalence of nonnegative matrix factorization and spectral clustering, in: SIAM International Conference on Data Mining, 2005.

[42] D. Hunter, K. Lange, A tutorial on MM algorithms, Am. Stat. 58 (1) (2004) 30–37.

[43] Z. He, S. Xie, R. Zdunek, G. Zhou, A. Cichocki, Symmetric nonnegative matrix factorization: Algorithms and applications to probabilistic clustering, IEEE Trans. Neural Networks 22 (12) (2011) 2117–2131.

[44] A. Lancichinetti, S. Fortunato, Community detection algorithms: A comparative analysis, Phys. Rev. E 80 (2009) 5.

[45] D. Gomez, J.T. Rodrguez, J. Yanez, J. Montero, A new modularity measure for Fuzzy Community detection problems based on overlap and grouping functions, Int. J. Approximate Reasoning 74 (2016) 88–107.

[46] C. Lee, F. Reid F, A. McDaid and N. Hurley N, Detecting highly overlapping community structure by greedy clique expansion, arXiv:1002.1827, 2010.

[47] X. Bai, P. Yang, X. Shi, An overlapping community detection algorithm based on density peaks, Neurocomputing 226 (2017) 7–15.

[48] H. Du, W. Wang, L. Bai, An overlapping community detection algorithm based on centrality measurement of network node, J. Comp. Res. Dev. 55 (8) (2018) 1619–1630.

[49] Y.-Y. Ahn, J.P. Bagrow, S. Lehmann, Link communities reveal multiscale complexity in networks, Nature 466 (7307) (2010) 761–764.

[50] A. Lancichinetti, S. Fortunato, F. Radicchi, Benchmark graphs for testing community detection algorithms, Phys. Rev. E 78 (2008) 046110.

[51] C. Gaiteri, M. Chen, B. Szymanski, K. Kuzmin, J. Xie, C. Lee, T. Blanche, E.C. Neto, S.-C. Huang, T. Grabowski, et al., Identifying robust communities and multi-community nodes by combining top-down and bottom-up approaches to clustering, Sci. Rep. 5 (1) (2015) 1–14.

[52] M. Brutz, F.G. Meyer, A flexible multiscale approach to overlapping community detection, Soc. Netw. Anal. Min. 5 (1) (2015) 1–17.

[53] A. Clauset, M.E. Newman, C. Moore, Finding community structure in very large networks, Phys. Rev. E 70 (6) (2004) 066111.

[54] M. Ebrahimi, M.R. Shahmoradi, Z. Heshmati, M. Salehi, A novel method for overlapping community detection using multi-objective optimization, Phys. A 505 (2018) 825–835.

Changjian Fang received the Ph.D. degrees from School of Cybersecurity of Northwestern Polytechnical University. Now he is currently an associate professor in School of Information Engineering, Nanjing Audit University. His current research interests include pattern recognition, text mining, social network analysis.

Zhen-Zhou Lin received the Master degrees from School of Management of Nanjing University of Posts and Telecommunications. Now he is currently the director of the Experimental Teaching Center of Economics and Management of Nanjing University of Finance and Economics. His current research interests include data mining, pattern recognition and complex network.