ModMRF: A Modularity-based Markov Random Field Method for Community Detection

Dongxiao He[a,*], Yue Song[a], Zhiyong Feng[b], Shizhan Chen[a], Weihao Li[c], Katarzyna Musial[d]

[a]*School of Computer Science and Technology, Tianjin University, Tianjin 300350, China*
[b]*School of Computer Software, Tianjin University, Tianjin 300350, China*
[c]*Visual Learning Lab, Heidelberg University, Heidelberg 69120, Germany*
[d]*Advanced Analytics Institute, School of Computer Science, Faculty of Engineering and IT, University of Technology Sydney, Sydney 2007, Australia*

Abstract

Complex networks are widely used in the research of social and biological fields. Analyzing real community structure in networks is the key to the study of complex networks. Modularity optimization is one of the most popular techniques in community detection. However, due to its greedy characteristic, it leads to a large number of incorrect partitions and more communities than in reality. Existing methods use the modularity as a Hamiltonian at the finite temperature to solve the **above** problem. Nevertheless, modularity is not formalized as a statistical model in the method, which makes many statistical inference methods limited and cannot be used. Moreover, the method uses the sum-product version of belief propagation (BP) which has not better performance than its max-sum version, since it calculates per-variable marginal probabilities rather than the joint probability. To address these issues, we propose a novel Markov Random Field (MRF) method by formalizing modularity as an energy function based on the rich structures of MRF to represent properties and constraints of this problem, and use the max-sum BP to infer model parameters. In order to analyze our method and compare it with existing methods, we conducted experiments on both real-world and synthetic networks with ground-truth of communities, showing that the new method outperforms the state-of-the-art methods.

Keywords: Complex Networks; Community Detection; Overfitting; Belief Propagation; Modularity; Markov Random Field.

1. Introduction

Many complex systems in different fields (e.g., social science, genetic science, and information science) are generally abstracted as networks, where nodes represent elements, and edges represent mutual interactions between elements in the system. One of the significant property of the network is community detection, which refers to the aggregation of nodes in the network into communities. Generally, nodes within a community are densely connected, while connections of

*Corresponding author
Email addresses: hedongxiao@tju.edu.cn (Dongxiao He), sy717389667@tju.edu.cn (Yue Song), zyfeng@tju.edu.cn (Zhiyong Feng), zyfeng@tju.edu.cn (Shizhan Chen), zyfeng@tju.edu.cn (Weihao Li), zyfeng@tju.edu.cn (Kaska Musial-Gabrys)

nodes between communities are sparse. Detecting communities of a network can help us discover objects with the same function in the system, study the relationship between different communities, infer the attribute values missing from the objects, predict the relationships that have not been discovered between the objects, and so on [19]. Community detection has been successfully used in many applications, e.g., terrorist identification [36], behavior prediction [37], and recommendation system [38].

Researchers have conducted extensive research on community detection, resulting in a large number of community detection algorithms based on various assumptions and techniques, as reviewed in [1]. They include hierarchical clustering [27], spectral clustering [28], modularity optimization [2], Markov dynamic algorithms [29], and statistical model based methods [30].

Among these methods, we would like to highlight the optimization of network modularity (i.e., a function that quantifies the pros and cons of a community partition by comparing the difference between the edge density of each community in the real network and the corresponding subgraph in the random network [3]), which is to find community structures with the largest modularity. Though this method has been actively studying, it still suffers from some problems. Maximizing modularity often leads to overfitting, that is, the number of communities divided from a network is often larger than the real number of communities and more nodes are incorrectly allocated than existing methods (e.g., Mod [13]) Furthermore, there are many local optimal solutions for modularity in real networks, while there is no clear correlation between them [10].

Fortunately, Zhang *et al* [13] take network modularity as a Hamiltonian at the finite temperature, use belief propagation (BP) for inference, and give a principled way to determine the number of communities. Their method can obtain the consensus of many community partitions with high modularity values rather than looking for a single partition which maximizes modularity, so that it is possible to avoid these above drawbacks. However, Zhang *et al* did not formalize modularity as a statistical model, which makes their method limited to using BP alone to perform optimization and not free to various types of statistical inference methods that may be more powerful. At the same time, the special BP algorithm they used (the sum-product version) is only to compute the marginal probabilities of the joint probability distribution. However, the calculation of the largest joint probability directly is often believed having better performance than the optimization of individual per-variable marginal probabilities [19].

This paper is to solve these problems. We first formalize network modularity as a pairwise Markov Random Field (pMRF) via reparameterization of pairwise potentials using Gibbs distribution. In this way, modularity can be optimized by some different types statistical inference methods (e.g., belief propagation with both the sum-product and max-sum visions, variable elimination and MCMC), since pMRF is a typical undirected probabilistic graphical model. We then give a BP algorithm with the max-sum vision, which is to maximize the joint probability of community memberships (rather than maximizing the per-variable marginal probabilities as other work done [13]) for model inference. At this time, this above method is still not satisfactory since the pMRF model uses fully-connected pairwises and the calculation of joint probability is also time consuming. So, we further improve the mechanism of message propagation in BP by using the statistical properties of networks to introduce an external field [15, 20], which can reduce its time complexity from cubic to linear in the case of having certifiable results.

The rest of the paper is organized as follows. A brief review of the related work is given in Section 2. In Section 3, we first give the related preliminary knowledge and then present the model and inference as well as its speedup method. The experimental results and analysis are described in Section 4. We conclude this paper with some highlights in Section 5.

2. Related Work

Our work is inspired by the recent work on modularity and MRF for community detection and image processing respectively.

2.1. Modularity Optimization for Community Detection

Network modularity, proposed by Newman and Girvan [3], is applied to measure the significance of community structures. Since the definition of modularity was given, several optimization strategies for modularity have been proposed [4] for community detection. They include greedy algorithms [2, 11, 12], simulated annealing [7], extremal optimization [8], spectral clustering [9] etc. Take Louvain method [2] as an example, it is often believed to run very fast and have good performance. However, these methods are more inclined to find coarse community structures than precise ones, and suffer from overfitting. That is, they produce many illusory communities in the networks which often do not exist. Although some variants of modularity that improved the accuracy have been proposed [23, 24, 25], the overfitting has been still not solved and the generalization of the models needs to be further strengthened.

In order to solve the problems caused by maximizing modularity, Zhang *et al* [13] propose to use modularity as a Hamiltonian at the finite temperature and propose a new method to calculate the number of communities. Thereafter, they introduce Gibbs distribution as a function of inverse temperature. In model inference, they give a belief propagation (BP) algorithm to approximate the marginal probabilities. Although this method has greatly improved the accuracy of community partitions and avoids overfitting, they overlook two important problems. First, they did not formalize modularity as a type of statistical models, which makes that many statistical inference methods cannot be freely used in learning. Second, they did not consider the difference between the real joint probability distribution and its marginal probabilities approximation. That is, the marginal probabilities may lead to that some useful information is not properly utilized, resulting in dissatisfactory community partitions; while the joint probability distribution is often expected to provide better solutions than the optimization of individual per-variable marginal probabilities as they used.

2.2. Markov Random Field for Image Segmentation

Markov Random Field (MRF), as a type of undirected probabilistic graphical model, has achieved great success in image processing and computer vision [14]. The objective function of MRF model is generally defined as an energy form, which is the core of the model, and then the energy function is subsequently transformed to a probabilistic objective function by using a Gibbs distribution. Some popular inference methods such as belief propagation, mean field, and Monte Carlo approaches can be used to learn the MRF models. Here, we are particularly interested in pairwise MRF (pMRF), which is one of the most popular type of MRFs in image segmentation [26, 31, 32]. Energy function of the pMRF model often consists of unary potentials and pairwise potentials, which are defined by pixel features and adjacency relationships respectively. For example, Krhenbhl and Koltun [33] apply the fully connected pairwise MRF model to deal with the image segmentation problem, where the unary potentials used in their implementation take shape, texture, location, and color descriptors into consideration, and pairwise potentials are defined by a linear combination of Gaussian kernels in an arbitrary feature space. Here we wish to formalize network modularity as an energy function of pMRF which has been not considered by previous works.

3. The Method

We first introduce some notations and preliminaries. We then present 1) how to formalizes network modularity as pMRF energies, 2) how to perform model inference by maximizing the joint probability distribution, and 3) how to further speed up the inference by using external field theory. We finally give its complexity analysis.

3.1. Preliminaries

The notations and the definition of the problem of community detection are first introduced. We then discuss the definition of modularity and pMRF respectively.

3.1.1. Notations and the Problem

Consider an undirected and unweighted network $G = (V, E)$ with n nodes and m edges, where V denotes all nodes in the network G and E denotes the edge set. We use a $n \times n$ adjacent matrix A to denote whether there is an edge between nodes. That is, if nodes i and j are connected, A_{ij} is 1 or 0 otherwise. $C = \{c_1, c_2, c_1, ..., c_n\}$ is used to represent a community partition, where c_i denotes to which community node i belongs. The task of community detection is to divide n nodes into K communities.

3.1.2. Network Modularity

Network modularity which is a mostly used quality measure of community structures has been proposed by Newman and Girvan in [3]. It is defined as the fraction of edges that the number of edges between nodes within communities minus the expected value of the same quantity in a network with the same community partitions but edges are assigned at random, conditional on the given community memberships and the degrees of nodes. More formally, let c_i be the community to which node i is assigned. Modularity Q of the community partition is defined as:

$$Q(C) = \frac{1}{m} \sum_{i,j \in V} \left(A_{ij} - \frac{d_i d_j}{2m} \right) \delta \left(c_i, c_j \right) \tag{1}$$

where d_i denotes the degree of node i, and $\delta \left(c_i, c_j \right)$ is the Kronecker delta, which is 1, if $c_i = c_j$, or 0 otherwise. If Q equals to 0, the community structure derived is almost random. But if Q approaches the maximum, which is 1, we have a very strong community structure.

3.1.3. Pairwise Markov Random Field

Pairwise Markov Random Field (pMRF) has been typically used in image segmentation. For example, let n be the set of pixels in an image, $X = (X_i)_{i \in n}$ the random field of classes, $Y = (Y_i)_{i \in n}$ the random field of observations, K the number of classes, and C a set of variable cliques which are defined by the neighborhood system. Then, the idea of pMRF models is to consider directly the Markovianity of the pairwise random field $Z = (X, Y)$. The energy function is composed by a set of unary potentials $\sum_i \theta_i$, which measures the cost that nodes have labels, and a set of pairwise potentials $\sum_{ij,(i,j) \in C} \theta_{ij}$, which represents the sum of the cost that node pairs have labels, respectively, as showed in Fig. 1. The distribution of a general case of pMRF can then be written as:

$$P(X = x, Y = y) = \lambda exp \left(- \sum_i \theta_i - \sum_{ij,(i,j) \in C} \theta_{ij} \right) \tag{2}$$

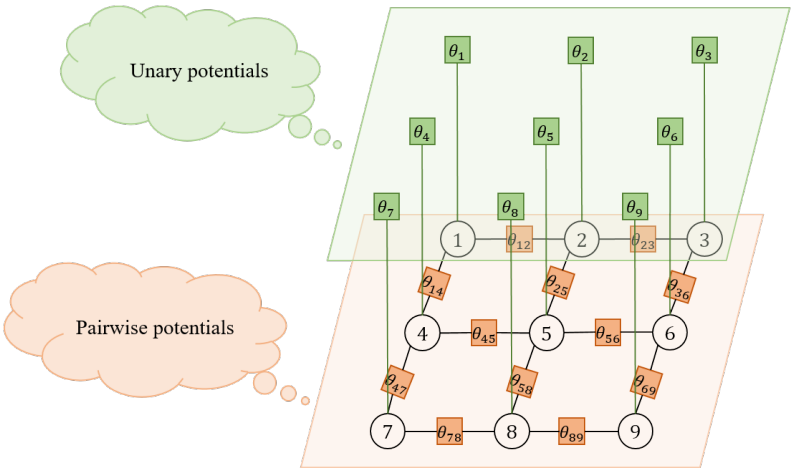


Figure 1: Graphical representation of a typical pMRF model. The black circles denote pixels. $\{\theta_1, ..., \theta_9\}$ and $\{\theta_{12}, ..., \theta_{89}\}$ denote the set of unary potentials and pairwise potentials, respectively.

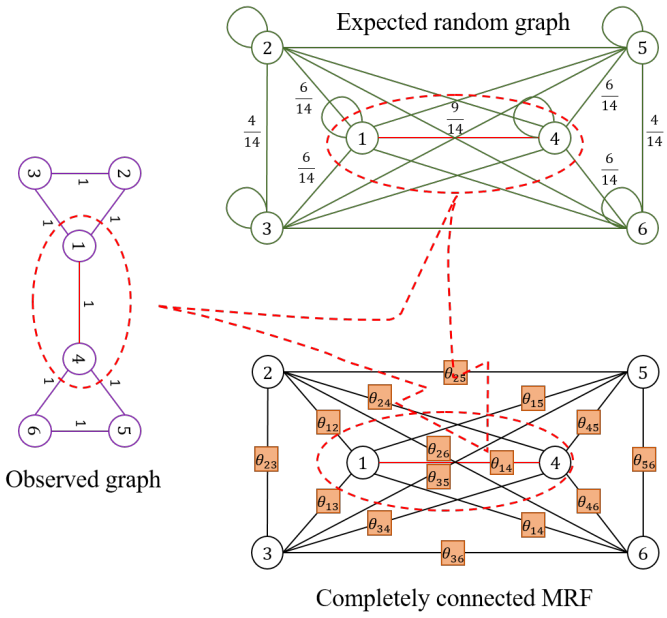


Figure 2: Graphical representation of the proposed model in network community detection. The black, green and purple circles denote nodes. $\{\theta_{12}, ..., \theta_{56}\}$ denotes the set of pairwise potentials. The data in the purple part and the green part denote the real number and expected number of edges between nodes, which are only partially shown for illustrating. The red dashed line represents the construction process of pairwise potentials.

3.2. Formalization of Modularity as MRF

We start to formalize modularity as an energy function of MRF, and then propose an efficient model inference algorithm and finish with a complexity analysis.

3.2.1. The Model

Our proposed method falls into pMRF with modularity so that it is termed as ModMRF for short. The core of this method is how to transform the modularity into the energy function of MRF. By doing so, more inference algorithms can be used to calculate community partitions such as the max-sum version of BP.

To be specific, first, the energy function of a pMRF model is the sum of pairwise potentials for all node pairs of a given network due to the lack of node feature information for building unary potentials and there is the property that the minimum of the energy function corresponds to the best possible community partition [26]. Meanwhile, the pairwise potential between nodes i and j, i.e., θ_{ij}, should represent the cost that they have labels c_i and c_j. Second, modularity is the sum of the difference between the real ratio of each edge between a pair of nodes within communities in the given network (the purple part in Fig. 2) and the expected one in a random graph (the green part in Fig. 2). The larger the difference, the larger the possibility that the two nodes are in a community will be. Then its negative value corresponds to that: the smaller the pairwise potential between two nodes, the smaller the cost of having labels c_i and c_j will be. Therefore, the pairwise potentials can be defined based on the relative density of edges between each pair of nodes. So the pairwise potential between nodes i and j can then be defined as (the red dashed line part in Fig. 2):

$$\theta_{ij}\left(c_i, c_j; A_{ij}\right) = -\left(A_{ij} - \frac{d_i d_j}{2m}\right)\delta\left(c_i, c_j\right) \tag{3}$$

where the definitions of symbols are the same as that in (1). The energy function with exclusively the pairwise potentials of our model can then be defined as:

$$E(C) = \sum_{i,j\in V} \theta_{ij}\left(c_i, c_j; A_{ij}\right) = \sum_{i,j\in V} -\left(A_{ij} - \frac{d_i d_j}{2m}\right)\delta\left(c_i, c_j\right) \tag{4}$$

It satisfies that the larger the modularity is, the smaller the energy function is, and thus the more likely the community partition will be.

Then Gibbs distribution is introduced as a function of inverse temperature β:

$$P(C) \propto exp\left\{-\beta E(C)\right\} \tag{5}$$

Therefore, the joint probability that each node belongs to which community in the entire network can be expressed as:

$$P(C) = \frac{1}{Z} exp\left\{\sum_{i,j\in V} \beta\left(A_{ij} - \frac{d_i d_j}{2m}\right)\delta\left(c_i, c_j\right)\right\} \tag{6}$$

where Z is a partitioning function, which depends on the adjacency matrix $\left[A_{ij}\right]_{n\times n}$ and helps guarantee that $P(C)$ satisfies the definition of a probability distribution. Finally, we can get a

globally optimal community partition by maximizing:

$$\begin{aligned} \overset{*}{C} &= \arg\max_{C} \; P(C) \\ &= \arg \max_{c_1} \ldots \max_{c_n} P(C) \end{aligned} \tag{7}$$

3.2.2. Max-Sum BP for Model Inference

After network modularity is formalized into the energy function of MRF, a series of inference algorithms can be used to learn the model. In order to obtain the community partition C that maximizes the probability distribution $P(C)$ of the MRF model, we adopt the max-sum version of BP algorithm to find a configuration of community memberships of all nodes. This algorithm finds a configuration of variables that has the largest joint probability, which is often believed having better performance than the sum-product version of BP that computes marginal probabilities of the joint probability distribution [19]. For instance, as shown in Table 1, the marginal probabilities of nodes A and B both belonging to community 1, which are the maximum and are 0.6 and 0.7, respectively, corresponding the joint probability value 0.3. But this is smaller than the real maximization of joint probability that node A belongs to community 2 and node B belongs to community 1, which is 0.4. This is because the joint probability considers the hidden relationship between communities, and hence it will be more reasonable to obtain community partitions by maximizing joint probability rather than maximizing marginal probabilities (as did in [34]).

Table 1: Probability distribution of communities of two nodes in a toy network.

Node B's label \ Node A's label	Community 1	Community 2	Marginal prob of B
Community 1	0.3	**0.4**	0.7
Community 2	0.3	0.0	0.3
Marginal prob of A	0.6	0.4	

We now introduce how to use the max-sum version of BP algorithm to derive the optimal community structure in the model. First, this method collects messages that are passed by all nodes except the node for which the calculations are performed, and the messages are passed along the edges. If this is a ring network, a certain number of iterations are required until the final messages and beliefs reach convergence. Because each node has more than one state space, the message is a set of vectors as the same dimension as the number of states. Here the number of states of the nodes is the total number of communities K. $m^{T}_{i\to j}(c_i)$ indicates the message sent by node i to its neighbor node j at a certain time T. The new message for each iteration is calculated as:

$$m^{T}_{i\to j}(c_i) \leftarrow \sum_{k\in \partial i\setminus j} \left\{ \max_{c_k} \left[\delta(c_i, c_k) \left(A_{ik} - \frac{d_i d_k}{2m} \right) \beta + m^{T-1}_{k\to i}(c_k) \right] \right\} \tag{8}$$

where ∂i is the set of neighboring nodes of node i. Note that the graph here is not the original network itself, but a fully connected graph (the black part in Fig. 2). So ∂i is the set of all nodes in the network except i. Messages are normalized in every iteration by moving the minimum of the message for any community state $c_i \in \{1, ..., K\}$ to zero, according to [26]. Then, the variable

beliefs in different states can be calculated as:

$$b_i(c_i) \leftarrow \sum_{k \neq i} \left\{ \max_{c_k} \left[\delta(c_i, c_k) \left(A_{ik} - \frac{d_i d_k}{2m} \right) \beta + m_{k \to i}(c_k) \right] \right\} \tag{9}$$

This belief can be taken as a scoring function to find the community to which the node most likely belongs to. At this point, the community label of node i (i.e., c_i) can be selected based on the nodes max-belief $b_i(c_i)$, i.e.,

$$\overset{*}{c}_i = \arg \max_{c_i \in \{1,\ldots,K\}} ... \max b_i(c_i) \tag{10}$$

The detailed description of the algorithm is as follows.

Algorithm 1 exact BP-inference.

Input: A network A_{ij}, *criterium*, t_max and q;
Output: Community partition $C = \{c_1, ..., c_n\}$;

Initialize the q-component normalized vectors $\left\{ m^0_{i \to j}(c_i) \right\}$ randomly;
$conv \leftarrow criterium + 10$; $t \leftarrow 0$;
while $conv > criterium$ and $t < t_{max}$: **do**
  $conv \leftarrow 0$; $t \leftarrow t + 1$;
  for every message $\left\{ m^t_{i \to j}(c_i) \right\}$ (in random order): **do**
    Update all q-components of $\left\{ m^{t+1}_{i \to j}(c_i) \right\}$
    $conv \leftarrow conv + \left| m^{t+1}_{i \to j}(c_i) - m^t_{i \to j}(c_i) \right|$
  end for
end while
Compute variable max belief $b_i(c_i)$ for each node i;
return group assignments $\overset{*}{C}$;

3.2.3. Speedup the Model Inference

Since our ModMRF model is a fully connected pairwise MRF, messages on all edges of the complete graph (the black part in Fig. 2) need to be computed in each iteration for a total of $n(n-1)$ messages, where n is the number of nodes. Also, according to (8) the time for computing each message is $O(n)$. So, the complexity of the exact algorithm above is $O(n^3)$, which is only suitable for small networks. However, the scale of real networks is usually much larger. Therefore, here we further speed up the above method. Based on the cavity theory in statistical physics [35], we can ultimately reduce the computational complexity to linear under the condition that remain the good performance.

To be specific, each node can then send a same message which equals to the belief of node i (i.e., $b_i(c_i)$) to all its non-neighbors of the original network, since the terms of subleading order can be neglected according to the cavity theory [15]. Thereafter, the messages sent to non-neighbors can be replaced by an external field [15]. In this case, in each iteration we only need to update $O(m)$ messages where m is the number of edges. Then the message that node i sends

to its neighbor j can be computed as:

$$\begin{aligned} m_{i\to j}^{T}(c_i) \leftarrow & \sum_{k\in\partial i}\left\{\max_{c_k}\left[\delta(c_i,c_k)\left(1-\frac{d_i d_k}{2m}\right)\beta + m_{k\to i}^{T-1}(c_k)\right]\right\} \\ & + \sum_{k\notin\partial i\setminus j}\left\{\max_{c_k}\left[\left(-\beta\frac{d_i d_k}{2m}\right)\delta(c_i,c_k) + m_{k\to i}^{T-1}(c_k)\right]\right\} \\ & \approx b_i(c_i) + O\left(\frac{1}{N}\right) \end{aligned} \tag{11}$$

where ∂i is the set of neighboring nodes of node i in the observed network. Therefore, the messages from non-neighbors can be calculated based on auxiliary external field as follows:

$$m_{i\to j}^{T}(c_i) \leftarrow \sum_{k\in\partial i}\left\{\max_{c_k}\left[\delta(c_i,c_k)\left(1-\frac{d_i d_k}{2m}\right)\beta + m_{k\to i}^{T-1}(c_k)\right]\right\} + H(D) \tag{12}$$

where ∂i is the neighborhood of node i on network G, and $H(D)$ is the external field, that is:

$$H(D) = \sum_{k}^{N}\left\{\max_{c_k}\left[\left(-\beta\frac{D d_k}{2m}\right)\delta\left(c_i,c_j\right) + b_k(c_k)\right]\right\} \tag{13}$$

where D are the distinct degrees. Assuming that L is the number of the distinct degrees, there are L external fields that need to be calculated in each iteration, which are brought into equation (12). Then, the belief can be calculated based on the auxiliary external field according to:

$$b_i(c_i) \leftarrow \sum_{k\in\partial i}\left\{\max_{c_k}\left[\delta(c_i,c_k)\left(1-\frac{d_i d_k}{2m}\right)\beta + m_{k\to i}^{T}(c_k)\right]\right\} + H(D) \tag{14}$$

In order to find a fixed point of (12), when calculating the message $m_{i\to j}^{T}(c_i)$, the old value is subtracted from the new one, and then the belief $b_i(c_i)$ and the external field $H(D)$ can be updated respectively. Similarly, when there are loops in the network, the calculation will be repeated more times until convergence. The detail description of the whole algorithm is described as follows:

3.3. Complexity Analysis

The time complexity algorithm 1 is $O(n^3)$, where n is the number of nodes. After speeding up (i.e., algorithm 2), for each edge (i, j), we require $O(K^2 d_i)$ time to calculate the message $m_{i\to j}^{T}(c_i)$ and the belief $b_i(c_i)$ according to (12) and (14), respectively, where d_i is the degree of node i. Also, for the calculation of the external field according to (13), the complexity of each iteration is $O(K^2 L)$. Therefore, for the entire network system, the total time consumed for one iteration is $O(mK^2(d_{\max} + L))$, where $d_{\max}$ denotes the maximum value of degrees of all nodes. But due to the sparsity of large networks, $d_{\max}$ and L are very small relative to the number of nodes n. And also, the number of iterations is often taken as a constant (such as 100 as used in general). So, the time complexity of algorithm 2 is $O(m)$, which is nearly linear to large sparse networks.

4. Experimental Results and Analysis

Here we validate the effectiveness of our newly proposed MRF method by comparing several existing methods on some widely used benchmarks.

Algorithm 2 approximate BP-inference.

Input: A network A_{ij}, *criterium*, t_max and q;
Output: Community partition $C = \{c_1, ..., c_n\}$;

Initialize the q-component normalized vectors $\left\{m^0_{i\to j}(c_i)\right\}$ randomly;
Compute belief $b_i(c_i)$ for each node i;
Compute the q-component auxiliary field $H(c_i)$;
$conv \leftarrow criterium + 10$; $t \leftarrow 0$;
while $conv > criterium$ and $t < t_{max}$: **do**
  $conv \leftarrow 0$; $t \leftarrow t + 1$;
  for every message $\left\{m^t_{i\to j}(c_i)\right\}$ (in random order): **do**
    Update all q-components of $\left\{m^{t+1}_{i\to j}(c_i)\right\}$;
    $conv \leftarrow conv + \left|m^{t+1}_{i\to j}(c_i) - m^t_{i\to j}(c_i)\right|$;
    Update $b_i(c_i)$ using the new value of $\left\{m^{t+1}_{i\to j}(c_i)\right\}$;
    Update the external field H;
  end for
end while
Compute variable max belief $b_i(c_i)$ for each node i;
return group assignments $\overset{*}{C}$;

We apply our algorithm (i) first to 10 real-world networks and (ii) then to synthetic networks. In real-world datasets, we compare and analyze the approach we have designed with five baseline methods. They include FN [5], LEV [16], CNM [6], Louvain [2] and Mod [13]. The first four methods obtain the results of community partition by finding the largest modularity and, according to our knowledge, Louvain is one of the best and most widely used algorithms in the modularity optimization family. Mod was designed by Zhang *et al* [13] based on the modularity theory and is learned by the sum-product BP method. To further illustrate that our method has better performance than Mod, we conducted further analysis on synthetic networks.

4.1. Metrics and the Number of Communities

We present the metrics used to verify the performance of our method and the determination of the number of communities.

4.1.1. Metrics

We will use three different metrics to measure the quality of the community partition, namely modularity, NMI (normalized mutual information) and AC (accuracy), because all methods are based on the modularity theory and the ground-truth of communities is known. We use the E.q. (1) to calculate modularity Q.

NMI is commonly used to measure the similarity between the detected community structure and the real one in clustering approaches [18]. NMI is an important measurement used in the case of community detection with ground-truth. It can be used to objectively evaluate the accuracy of the results of a community partition and standard community results. The NMI value ranges from 0 to 1, and the higher the value, the closer the partition result to the exact answer, which

means the higher the accuracy of the partition. The formula of NMI is defined as:

$$I(A,B) = \frac{-2\sum_{i=1}^{C_A}\sum_{j=1}^{C_B} C_{ij}\log\left(C_{ij}N/C_iC_j\right)}{\sum_{i=1}^{C_A} C_i\log\left(C_i/N\right)+\sum_{j=1}^{C_B} C_j\log\left(C_j/N\right)} \tag{15}$$

where A and B denote two given community structures, and $[C_{ij}]_{|A|\times|B|}$is a scrambling matrix. C_{ij} denotes the members of community j in B and the same parts of community i in A. N is the number of nodes in the network. C_i is the sum of all the elements in the i-row of the matrix $[C_{ij}]_{|A|\times|B|}$. Similarly, C_j is the sum of the elements of the j-column in the matrix.

AC is used to measure the percentage of correct labels obtained by using above algorithms. Given a network with N nodes. Assume C_i is the community label we obtained by an above algorithm, and R_i is the label in the ground-truth. The accuracy AC is defined as:

$$AC = \frac{\sum_{i=1}^{N}\delta\left(R_i, map\left(C_i\right)\right)}{N} \tag{16}$$

where $\delta\left(R_i, map\left(C_i\right)\right)$ is the Kronecker delta, and $map\left(C_i\right)$ is the mapping function that maps each community label C_i to the equivalent label from the ground truth. The best mapping can be found by using the Kuhn-Munkres algorithm [22].

4.1.2. The Number of Communities

For choosing the number K of communities in a network, it is a classic model selection problem. Setting K by maximizing the modularity is a widely used heuristic in community detection, but it has been prone to overfitting. Zhang *et al* [13] provide a principled way to choose the number of communities, which uses the retrieval modularity as a criterion for choosing K, namely, for larger K, the retrieval phase becomes narrower, and the retrieval modularity does not increase [13], unlike other algorithms that tend to overfit. Thus, we also determine the value of K by introducing the way proposed by Zhang *et al*, rather than overfitting. Although our method is able to recognize how many communities there is, in order to compare fairly with the Mod method, we set K as that of Mod method, which is learned by adopting the way proposed by Zhang *et al* [13].

4.2. Real-World Networks

In this section, we first briefly describe the real-world networks used. Then on the real-world networks, FN, LEV, CNM, Louvain, Mod and the proposed approach ModMRF were compared and analyzed in terms of the above-mentioned three metrics (i.e. modularity, NMI and AC).

4.2.1. Datasets Description

In Table 2, a total of 10 real-world networks are listed. The scale of the datasets ranges from dozens of nodes to tens of thousands of nodes, and even the maximum number of edges can reach hundreds of thousands. More detailed analysis of these networks can be found in [17, 21, 23]. In the next demonstration of the experimental results, we will represent these networks with the abbreviation of the datasets.

Table 2: Datasets descriptions of ten real-world networks. n is the number of nodes, m the number of edges, and c the number of communities. 'High school friendship6' and 'High school friendship7' have the same network with different "true" community partitions.

Datasets' name	n	m	c	Abbreviation
Zacharys Karate club	34	78	2	Karate
High school friendship6	69	220	6	School6
High school friendship7	69	220	7	School7
Political books	105	441	3	Polbooks
American college football	115	613	12	Football
Cora	2,708	5,429	7	Cora
Citeseer	3,312	4,732	6	Cite
UAI2010	3,363	45,006	19	Uai2010
Northeastern	13,882	381,935	7	Northeastern
PubMed diabetes	19,729	44,338	3	Pubmed

Table 3: Comparison of 6 methods in terms of modularity on 10 real networks. Bold figure means better performance. The '/-/' denotes run time >48 hours.

Datasets	Modularity					
	FN	LEV	CNM	Louvain	Mod	ModMRF
Karate	0.372	0.393	0.381	0.419	0.371	**0.422**
Friendship6	0.585	0.544	0.545	0.593	0.597	**0.614**
Friendship7	0.571	0.544	0.545	0.593	0.597	**0.614**
Polbooks	0.501	0.467	0.502	0.52	0.521	**0.533**
Football	0.507	0.493	0.55	**0.605**	0.572	0.586
Cora	0.758	0.741	0.806	**0.813**	0.759	0.77
Cite	0.783	0.875	0.892	**0.9**	0.786	0.788
Uai2010	/-/	0.35	0.394	**0.461**	0.435	0.437
Northeastern	/-/	0.382	0.411	**0.506**	0.496	0.498
Pubmed	/-/	0.657	0.728	**0.77**	0.748	0.722

4.2.2. Quantitative Analysis

To validate the performance of our method, we compared ModMRF with two types of the state-of-the-art community detection methods, which are modularity optimization based methods [2, 5, 6, 16] and modularity as a Hamiltonian [13] respectively. In the experiment, we used all the datasets from Table 2 and tested the six algorithms on each of the datasets. Each setting of the experiment was repeated 20 times. We report the result with the highest objective. The experimental results are shown in Tables 3, 4 and 5.

Based on the modularity, ModMRF is on average 5.78%, 8.99%, 3.84%, and 1.70% more accurate than FN, LEV, CNM, and Mod, respectively. As shown in Table 3, except Louvain, ModMRF performs better than the other four methods on most of the datasets. Although some other approaches are superior to ModMRF in Cora and Cite datasets, modularity only tends to evaluate the clarity of community structure (i.e., the larger the modularity, the tighter connections within the community and the sparser connections between communities), and whether the communities to which nodes are correctly classified cannot be precisely judged. In addition, due to that we adopt the real-world networks with ground-truths of communities, the comparison is

Table 4: Comparison of 6 methods in terms of NMI on 10 real networks. Bold figure means better performance. The '/-/' denotes run time >48 hours.

Datasets	NMI					
	FN	LEV	CNM	Louvain	Mod	ModMRF
Karate	0.837	0.677	0.692	0.587	**1**	**1**
Friendship6	0.721	0.843	0.735	0.852	0.93	**0.961**
Friendship7	0.744	0.865	0.742	0.878	0.887	**0.916**
Polbooks	0.534	0.52	0.531	0.512	**0.542**	0.537
Football	0.673	0.699	0.698	**0.89**	0.71	0.718
Cora	0.374	0.421	0.455	**0.456**	0.33	0.384
Cite	0.109	0.331	**0.34**	0.337	0.18	0.146
Uai2010	/-/	0.118	0.147	**0.192**	0.152	0.152
Northeastern	/-/	0.303	0.427	0.473	0.549	**0.55**
Pubmed	/-/	0.18	**0.22**	0.204	0.183	0.172

Table 5: Comparison of 6 methods in terms of modularity on 10 real networks. Bold figure means better performance. The '/-/' denotes run time >48 hours.

Datasets	AC					
	FN	LEV	CNM	Louvain	Mod	ModMRF
Karate	0.971	0.618	0.735	0.647	**1**	**1**
Friendship6	0.667	0.768	0.71	0.826	0.928	**0.942**
Friendship7	0.739	0.826	0.725	**0.913**	0.841	0.855
Polbooks	0.819	0.714	0.81	0.724	**0.829**	0.819
Football	0.522	0.626	0.574	**0.87**	0.487	0.504
Cora	0.53	0.422	0.396	0.368	0.484	**0.547**
Cite	0.312	0.187	0.191	0.185	0.301	**0.315**
Uai2010	/-/	0.233	0.192	0.265	0.251	**0.267**
Northeastern	/-/	0.442	0.551	0.647	**0.694**	0.69
Pubmed	/-/	0.291	**0.413**	0.205	0.275	0.277

more compelling. So, the gold metric, i.e., accuracy, can be also used to evaluate the performance of different methods. Accuracy can more precisely assess the quality of community partition.

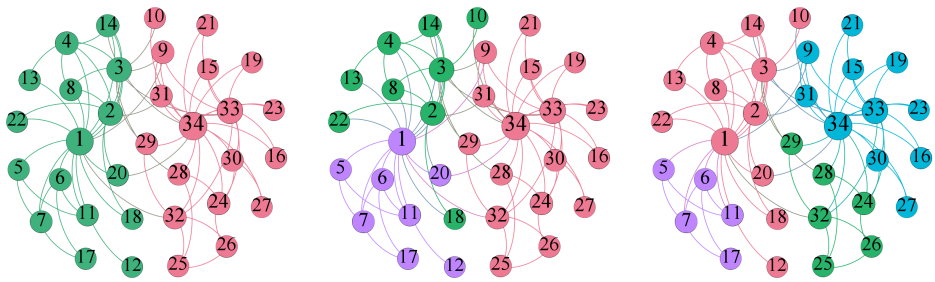
Tables 4 and 5 show the NMI and AC values of the six methods tested on the ten datasets. We observe that ModMRF has the best performance on 4 and 5 of the 10 networks in terms of NMI and AC, respectively. Using the NMI, ModMRF is on average 14.37%, 10.46%, 9.92%, 2.80% and 1.32% more accurate than FN, LEV, CNM, Louvain and Mod, respectively. We also obtained better results in terms of AC. Those are 8.47%, 17.52%, 14.78%, 9.11% and 2.03% for FN, LEV, CNM, Louvain and Mod, respectively. About 80% AC results of the first four methods are worse than those of ModMRF, although on the Cora, Cite and Uai2010 datasets, the modularity results (Table 3) e.g. for Louvain, are better than the ones for ModMRF. Comparing with the competitive Mod algorithm, the NMI and AC of ModMRF are better than that of Mod on 7 and 8 out of the 10 used datasets, respectively. These further validate the superiority of ModMRF over others in finding communities.

4.2.3. Qualitative Analysis

To further validate the performance of our method, here we first show the number of communities learned by LEV, CNM and Louvain methods on Cora and Cite datasets due to their better performance than ModMRF in terms of modularity, and then illustrate the number of communities in the test datasets of Karate which were determined by CNM, Louvain, Mod and our method, respectively, because they are all considered to be the more advanced algorithms.

First, we observe that the real number of communities on the Cora and Cite datasets are 7 and 6 groups respectively. But the number of communities extracted by LEV, CNM and Louvain methods are 106 and 392, 105 and 398, and 102 and 392 on the Cora and Cite datasets, respectively. This is because they use the method of maximizing modularity to discover the communities in the network, so that the network is partitioned into too many very small communities. The number of communities obtained by our method are 7 and 8 on Cora and Cite respectively, which is almost the same as the real number of communities. By the above experimental results, we can observe that modularity is biased as a measure in fact.

Furthermore, we visualize the community partitions on Karate club network given by CNM, Louvain, Mod and ModMRF. Since Mod and our methods partition results are the same as the real network partition, they are shown in Fig. 3 (a). Each node in the graph represents a member in the club, including the coach (node 1) and the administrator (node 34). Different colors represent different communities. The size of the node indicates the size of the degree of the node. As shown in (a), our method accurately divides the karate dataset into two communities that are centered on nodes 1 and 34, respectively, while the CNM and Louvain algorithms divided the network into 3 and 4 clubs, respectively, because of maximizing the modularity. Therefore, our method is superior to the method based on modularity optimization in terms of performance.



(a) (b) (c)

Figure 3: Comparison of 4 methods, including CNM, Louvain, Mod and the proposed method ModMRF in terms of the real community partition on the Zachary's karate club dataset. (a) For the real network partition result, Mod and our algorithm's results are the same, for a total of 2 communities; (b) CNM's result, for a total of 3 communities; (c) Louvain's result, for a total of 4 communities.

4.3. Synthetic Networks

Mod method is competitive with our method ModMRF, which can be seen from the experiments of the real datasets. In order to more clearly and intuitively see the difference between the these two algorithms, we use the currently accepted LFR benchmark network with known community structures as the artificial datasets to further evaluate the performance of our method regarding modularity, NMI and AC.

4.3.1. Experimental setup

LFR was proposed by Lancichinetti [18], which considers the heterogeneity of the real networks. So the LFR benchmark network is considered to be the closest to a real-world network. We designed two groups of experiments by changing different parameters on the benchmark network, and then used modularity, NMI and AC to evaluate the performance of our method. Some important parameters of the network are shown in Table 6.

Table 6: Parameters of the LFR benchmark networks

Symbols	Descriptions
N	number of vertices
$< d >$	average degree
d_{max}	maximum degree
α	exponent of the degree distribution
β	exponent of the community size distribution
μ	mixing parameter
C_{min}	minimum community sizes
C_{max}	maximum community sizes

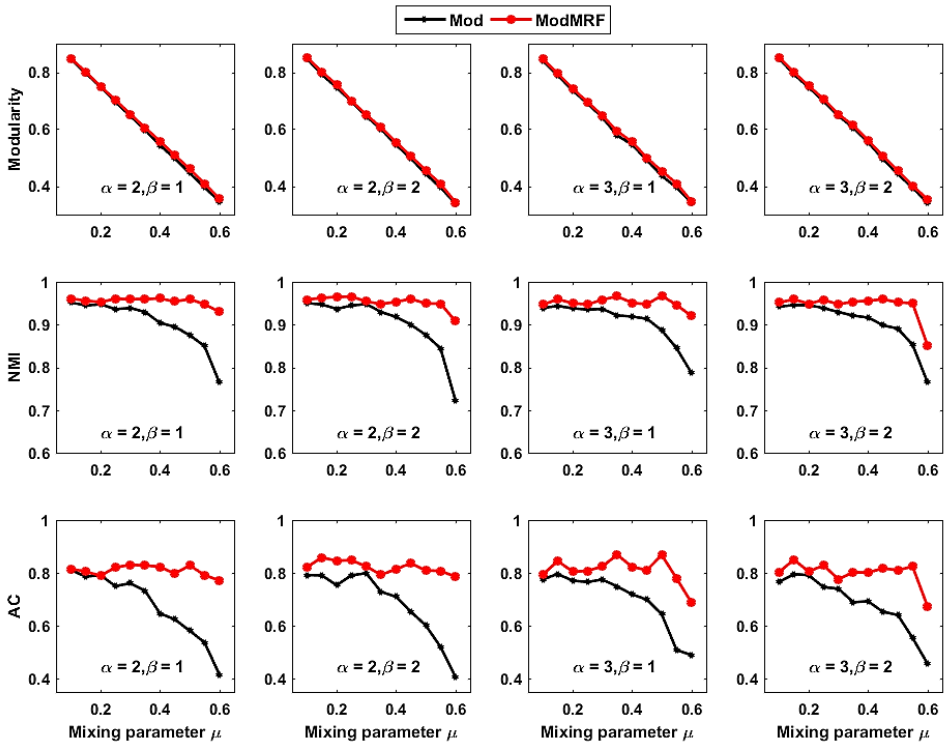


Figure 4: Comparison of Mod and ModMRF methods in terms of modularity, NMI and AC on LFR benchmark networks. The number of nodes $N = 1,000$. Each point corresponds to an average over 20 graph realizations.

First group: We design small networks with 1,000 nodes. We generated four pairs of graphs that correspond to 4 pairs of parameters $(\alpha, \beta) = (2,1), (2,2)(3,1)(3,2)$ respectively. Because there are the similar results when the average degree of the nodes has different values, it is set to 15 and the maximum degree is 50 (and $C_{min} = 20, C_{max} = 60$). We change the mixing parameter μ from 0.1 to 0.6 with an increment of 0.05. When the mixing parameter is larger, the community structure is harder to detect.

Second group: To check how the performance of our method is affected by the size of the network, we further compared our method with the Mod algorithm on a benchmark network with 5,000 nodes. We only changed the maximum degree of the node and the size of the largest community (the change of the two does not affect the shape of the results curve).

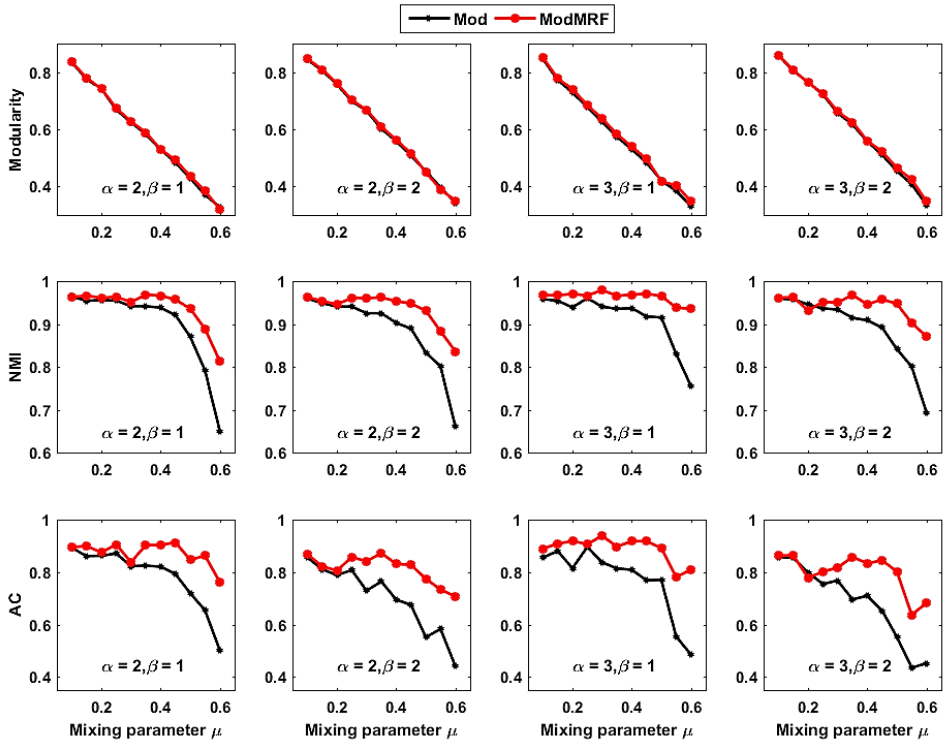


Figure 5: Comparison of Mod and ModMRF methods in terms of modularity, NMI and AC on LFR benchmark networks. The number of nodes $N = 5,000$, the other parameters are the same as those in Fig. 4. Each point corresponds to an average over 20 graph realizations.

4.3.2. Experimental Results

We first compared ModMRF with Mod method on the datasets with 1,000 nodes. As shown in Fig. 4, when the mixing parameter μ is close to 0, the community structure of the network is very obvious. The gap between the community structure detected by the two algorithms and the real community structure is very small. The NMI values of the two algorithms are almost all above 0.95 and the values of AC are about 0.8. However, with the increase of the parameter μ, the community structure gradually becomes insignificant. Although the modularity of Mod

is still close to our algorithm, the NMI and AC values are much smaller than the ModMRF we designed. It shows that the proposed algorithm ModMRF outperforms Mod approach in this artificial network.

We further compared our method with Mod on benchmark networks with 5,000 nodes. As shown in Fig. 5, the curves corresponding to the same parameters are similar to Fig. 4, while quickly drop to the bottom for larger networks, especially for the Mod algorithm. However, our method still performs better than the Mod algorithm. Especially, when the value of μ is in the range of 0.3 to 0.6, it further shows better performance of our method.

5. Conclusion

In this paper, we proposed a new MRF approach, namely ModMRF, to formalize modularity as the energy function for community detection in undirected static networks. Experimental results showed that our method offers better accuracy than the state-of-the-art algorithms. In addition, for the efficiency problem, ModMRF has reduced the time complexity to a nearly linear case. This level of time complexity in community detection algorithms can be qualified as very efficient. Finally, we hope that the study of this paper will further help to improve existing methods and provide useful information for designing new community detection methods.

References

[1] S. Fortunato and D. Hric. Community detection in networks: A user guide. Physics Reports, 659: 1-44, 2016.

[2] V.D. Blondel, J.-L. Guillaume, R. Lambiotte and E. Lefebvre. Fast unfolding of communities in large networks. Journal of Statistical Mechanics: Theory and Experiment, 2008(10):P10008, 2008.

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

[4] S. Fortunato. Community detection in graphs. 486(3-5): 75-174, 2010.

[5] M. E. J. Newman. Fast algorithm for detecting community structure in networks. Phys. Rev. E, 69(6): 066133, 2004.

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

[7] R. Guimera and L. A. N. Amaral. Functional cartography of complex metabolic networks. Nature, 433(7028): 895-900, 2005.

[8] J. Duch and A. Arenas. Community detection in complex networks using extremal optimization. Phys. Rev. E,72(2): 027104, 2005.

[9] C.K. Tsung, H. Ho, S. Chou, J. Lin and S. Lee. A Spectral Clustering Approach Based on Modularity Maximization for Community Detection Problem. Computer Symposium (ICS), International. IEEE, 12-17, 2016.

[10] D. Lusseau, K. Schneider, O. J. Boisseau, P. Haase, E. Slooten and S. M. Dawson. The bottlenose dolphin community of Doubtful Sound features a large proportion of long-lasting associations. Behav Ecol Sociobiol, 54(4): 396-405, 2003.

[11] X. Yan, C. Shalizi, J. E. Jensen, F. Krzakala, C. Moore, L. Zdeborov, P. Zhang and Y. Zhu. Model selection for degree-corrected block models. Stat. Mech., 2014(5): P05007, 2014.

[12] T. Martin, B. Ball, and M. E. J. Newman. Structural inference for uncertain networks. Physical review E, 93(1): 012306, 2016.

[13] P. Zhang and C. Moore. Scalable detection of statistically significant communities and hierarchies, using message passing for modularity. Proc. Natl. Acad. Sci., 111(51):18144-18149, 2014.

[14] S. Nowozin, and C. H. Lampert. Structured Learning and Prediction in Computer Vision. Foundations and Trends in Computer Graphics and Vision, 6(3-4): 185-365, 2011.

[15] A. Decelle, F. Krzakala, C .Moore and L. Zdeborov. Inference and phase transitions in the detection of modules in sparse networks. Phys Rev Lett, 107(6):065701, 2011.

[16] M. E. J. Newman. Finding community structure in networks using the eigenvectors of matrices. Phys. Rev. E, 74(3): 036104, 2006.

[17] M. E. J. Newman. Real-world network data in Newmans homepage, http://www-personal.umich.edu/ mejn/net-data/. 2017.
[18] W. Zachary. An information flow model for conflict and fission in small groups. Journal of anthropological research, 33(4): 452473, 1977.
[19] A. P. Streich, M. Frank. Multi-Assignment Clustering for Boolean Data. Journal of Machine Learning Research, 13(1): 459-489, 2012.
[20] M. Khademian, M. M. Homayounpour. Monaural multi-talker speech recognition using factorial speech processing models. Speech Communication, 98: 116, 2018.
[21] P. Sen, G. Namata, M. Bilgic, L. Getoor, B. Galligher and E. R. Tina. Collective classification in network data. AI Magazine 29(3): 93-106, 2008.
[22] L. Plummer and D. Lovaszandm. MATCHING THEORY, NORTHHOLLAND, 1986.
[23] M. Chen, K. Kuzmin, B.K. Szymanski. Community detection via maximization of modularity and its variants. IEEE Trans. Comput. Social Syst. 1 (1): 4665, 2014.
[24] J. Xiang, T. Hu, Y. Zhang, K. Hu, J. M. Li, X. K. Xu, C. C. Liu and S. Chen. Local modularity for community detection in complex networks. Physica A: Stat. Mech. Appl. 443: 451459, 2016.
[25] S. Zhang, H. Zhao. Normalized modularity optimization method for community identification with degree adjustment. Phys. Rev. E, 88 (5): 052802, 2013.
[26] S. Nowozin and C.H. Lampert. Structured Learning and Prediction in Computer Vision. Foundations and Trends in Computer Graphics and Vision, 6(3-4): 185-365, 2011.
[27] S. W. Jia, L. Gao, Y. Gao, J. Nastos, Y. J. Wang, X. D. Zhang, H. Y. Wang. Defining and identifying cograph communities in complex networks. New Journal of Physics 17(1): 013044, 2015.
[28] M. Fanuel, C. M. Alaiz, and J. A. K. Suykens. Magnetic eigenmaps for community detection in directed networks. Physical Review E, 95(2-1): 022302, 2017.
[29] D. Jin, B Yang, C. Baquero, D. Y. Liu, D. X. He and J. Liu. Markov random walk under constraint for discovering overlapping communities in complex networks. Journal of Statistical Mechanics Theory & Experiment, 5: P05031, 2011.
[30] Z. Chen, X. Li, and J. Bruna. Supervised Community Detection with Hierarchical Graph Neural Networks. arXiv preprint arXiv, 1705.08415, 2018.
[31] A. Blake, P. Kohli, and C. Rother. Markov Random Fields for Vision and Image Processing. The MIT Press, 2011.
[32] S. Zheng, S. Jayasumana, B. Romera-Paredes, V. Vineet, Z. Su, D. Du, C. Huang and P. Torr. Conditional Random Fields as Recurrent Neural Networks. IEEE International Conference on Computer Vision, 1529-1537, 2016.
[33] P. Krhenbhl,and V. Koltun. Efficient Inference in Fully Connected CRFs with Gaussian Edge Potentials. In The Twenty-fifth Annual Conference on Neural Information Processing Systems, 109-117, 2011
[34] C. M. Bishop. Pattern recognition and machine learning. springer, 2006.
[35] M. Zard and A. Montanari. Information, physics, and computation. Oxford University Press, 2009.
[36] F. Saidi and Z. Trabelsi and H. B. Ghazela. A novel approach for terrorist sub-communities detection based on constrained evidential clustering. 2018 12th International Conference on Research Challenges in Information Science, 1-8, 2018
[37] H. Yin, Z. Hu, X. Zhou, H. Wang, K. Zheng, Q. V. H. Nguyen, S. Sadiq. Discovering interpretable geo-social communities for user behavior prediction. 2016 IEEE 32nd International Conference on Data Engineering, 942-953, 2016.
[38] D. Bernardes, M. Diaby, R. Fournier, F. FogelmanSouli and E. Viennet. A Social Formalism and Survey for Recommender Systems, ACM SIGKDD Explorations Newsletter, 16(2): 20-37, 2015.