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

A Network-Specific Markov Random Field Approach to Community Detection

Dongxiao He,[1] Xinxin You,[2] Zhiyong Feng,[2] Di Jin,[1] Xue Yang[1,] Weixiong Zhang[3,4]

[1]School of Computer Science and Technology, Tianjin University, Tianjin 300072, China, [2]School of Computer Software, Tianjin University, Tianjin 300072, China, [3]College of Math and Computer Science, Institute for Systems Biology, Jianghan University, Wuhan 430056, China, [4]Department of Computer Science and Engineering, Washington University, St. Louis, MO 63130, USA
{hedongxiao, xinxyou, zyfeng, jindi}@tju.edu.cn, xsnowyang@163.com, weixiong.zhang@wustl.edu

Abstract

Markov Random Field (MRF) is a powerful framework for developing probabilistic models of complex problems. MRF models possess rich structures to represent properties and constraints of a problem. It has been successful on many application problems, particularly those of computer vision and image processing, where data are structured, e.g., pixels are organized on grids. The problem of identifying communities in networks, which is essential for network analysis, is in principle analogous to finding objects in images. It is surprising that MRF has not yet been explored for network community detection. It is challenging to apply MRF to network analysis problems where data are organized on graphs with irregular structures. Here we present a network-specific MRF approach to community detection. The new method effectively encodes the structural properties of an irregular network in an energy function (the core of an MRF model) so that the minimization of the function gives rise to the best community structures. We analyzed the new MRF-based method on several synthetic benchmarks and real-world networks, showing its superior performance over the state-of-the-art methods for community identification.

1. Introduction

We witnessed in recent years rapid accumulation of tremendous amounts of data from complex systems or networks, e.g., online social networks, World Wide Web, and biological networks. Finding network communities has become an effective means to study complex systems because highly connected nodes in the same community tend to share the same properties, and thus form functional modules. Examples of network communities include political groups reflected in blog networks and scientific disciplines in co-authorship networks. A fundamental problem for analyzing complex systems is identification of network communities or modules embedded in such large quantities of data. Discovering communities can help comprehend organizational principles of a system, understand its functions, and predict its future trend. Analysis of communities has been adopted in, e.g., customer segmentation based on their interests or preferences (Fortunato and Hric 2016), recommendation systems in social media consumption platforms (Bernardes et al. 2015), and behavioral social targeting (Pool, Bonchi, and Leeuwen 2014).

Detection of communities in complex networks was originally considered as a structure-based graph partition problem (Girvan and Newman 2002). Its primary objective is to divide the nodes in a network into groups so that the nodes within the same group are densely connected, whereas those in different groups are loosely connected. A wide variety of community detection algorithms have been proposed, as reviewed in (Fortunato and Hric 2016). They include hierarchical clustering (Girvan and Newman 2002), heuristic methods (Raghavan, Albert, and Kumara 2007), modularity-based methods (Newman and Girvan 2004), spectral algorithms (Li et al. 2015), dynamic algorithms (Rosvall and Bergstrom 2008), and statistical model based methods (Martin, Ball, and Newman 2016). Among these methods, statistical model based methods have a solid theoretical basis and reasonably good performance, and have been broadly adopted. Many statistical models have been explored and utilized to discover community structure, which include stochastic block model (Karrer and Newman 2011), nonnegative matrix factorization (NMF) (Jin et al. 2016), and deep learning (Yang et al. 2016).

Markov Random Field (MRF) is a general and potent statistical modeling technique (Nowozin and Lampert 2011). It can well represent structural relationships of task-specific properties and constraints underlying complex problems. It is particularly effective for problems in computer vision and image processing. However, to the best of our knowledge, MRF has not yet been considered for de-

Copyright © 2018, Association for the Advancement of Artificial Intelligence (www.aaai.org). All rights reserved.

tecting network communities, a special type of network structure. We like to highlight that finding communities in networks is analogous to finding objects or regions of interest in images (i.e., the image segmentation problem). The success of MRF on computer vision has inspired us to extend MRF to the problem of community detection.

Markov Random Field relates observations to a problem and some quantities of interest with a probabilistic model. The model encodes a joint probability distribution *P* over all feasible solutions of the quantities of interest, given the observations to the problem, and find an optimal solution by probabilistic inference on the model. To simplify an MRF model (without considering probabilistic constraints), the objective function of the model is first defined in an energy form (i.e., the well-known energy function for MRF) and the energy function is subsequently transformed to a probabilistic objective function (i.e., the joint *a posteriori* probability of the quantities of interest given the observations) by adopting a Gibbs distribution. Thus, the core of an MRF model is its energy function. An MRF model can be represented by an undirected graph, giving the alternative name of *undirected probabilistic graphical model*.

We are particularly interested in one of the most popular types of MRF for image segmentation, pairwise MRF (pMRF) (Nowozin and Lampert 2011). The energy function of a pMRF model is composed of a set of unary potentials and a set of pairwise potentials. For image segmentation, two types of information in images can be used. One is the set of features of pixels (e.g. color, brightness and texture) and the other is the set of spatial or adjacency relationships among pixels in an image. pMRF for image segmentation uses the pixel features to define a set of unary potentials and the adjacency relationships to define a set of pairwise potentials. In a pMRF model for image segmentation, pixel features (and unary potentials) play a dominant role by producing an initial segmentation of the image, and the spatial adjacency relationships of pixels (and pairwise potentials) are used to refine the model by removing noise using soft smoothing constraints.

Unfortunately, for the problem of finding communities in a network, network topology is often the only information available and no equivalent feature information is commonly provided. As a result, no feature information on nodes can be exploited in unary potentials, and we cannot derive an initial, rough community partition and directly adopt a pMRF model. Thus, in order to apply pMRF to the problem of network community identification only using network topology, we need to develop new pairwise potentials and a new network-specific pMRF approach.

We present in this paper such a pMRF model for network community identification. Different from a pMRF model for image segmentation that has one graph, our new network-specific pMRF model maintains three graphs: the original given network (adjacency relationships of nodes), an expected graph from a random-graph null model of the given network to help detect network properties, and an auxiliary, complete graph that is used as a graphical representation of the pMRF model. Furthermore, we further developed an efficient network-specific belief propagation (BP) method for model inference by utilizing network topological features. We present experimental results on synthetic benchmarks and real-world networks to show the superior performance of our network-specific MRF approach (named as NetMRF) over most state-of-the-arts.

2. The Net-MRF Approach

Given a network specified by the topology of how nodes are connected, finding communities in the network is equivalent to determining a community membership for every node. Here we present a novel approach using a MRF model that characterizes the relationship between the network topology and node community memberships. We also develop an inference method to optimize the model for finding the optimal community structure.

2.1 The NetMRF Model

2.1.1 Design of the Model

We focus on the case where no node feature is available for building unary potentials, so that the pMRF model only contains pairwise potentials for topological properties of a given network. Thus, making a full use of network structures holds the key to the design of a network-specific pMRF model. We introduce two innovative design ideas.

Although there is no precise definition of communities, one network structural property has been widely accepted and used, i.e., nodes in the same community are densely connected whereas nodes in different communities are loosely linked. Therefore, the relationship between two nodes with respect to an anticipated community structure can be in one of the following categories, i.e., they are connected (or disconnected) in the same community, or they are connected (or disconnected) in different communities. To fully utilize this relationship between two nodes, the pMRF model needs to 1) reward internal edges between nodes in the same group; 2) penalize the existing edges across different groups; 3) penalize missing edges (nonedges) between nodes in the same group; and 4) reward nonedges across different groups. These four requirements should be honored in a community detection method. Note that no existing method has considered these four requirements altogether. It is innovative to accommodate all of them in our network-specific MRF model.

A conventional pMRF model directly takes the adjacency relationships as the graphical representation of the model, and treats the adjacency relationships and the graphical representation of a model as the same. For the problem of

community detection, if node adjacency relationships (i.e., the network topology) are taken as a graphical representation of a pMRF model, the model only contains pairwise potentials for connected nodes in the given network. And it only realizes the first two of the four requirements described above and leaves out the last two requirements. To support all of the four requirements, we creatively treated the adjacency relationships and the graphical representation of a pMRF model separately and introduced an auxiliary, complete graph as the graphical representation to accommodate pairwise potentials for the last two requirements. This is the first innovation in our design.

We now turn to the intrinsic community structures that a given network has but a random graph does not. In a random graph, nodes are randomly connected so that the graph has little community structure. In contrast, real networks usually have some, albeit hidden, community structures. Thus, an expected random graph of the original network, which has the same number of nodes and the same number of edges as the given real network, can help extract information of community structures hidden in the given network. By comparing the real network with its expected random graph, we can assess the extent of denseness within communities and the extent of sparseness across communities with respect to the random case. Therefore, we introduced to our pMRF model an expected random graph of the original network as a baseline for comparison. This is the second innovation in our overall design of the model.

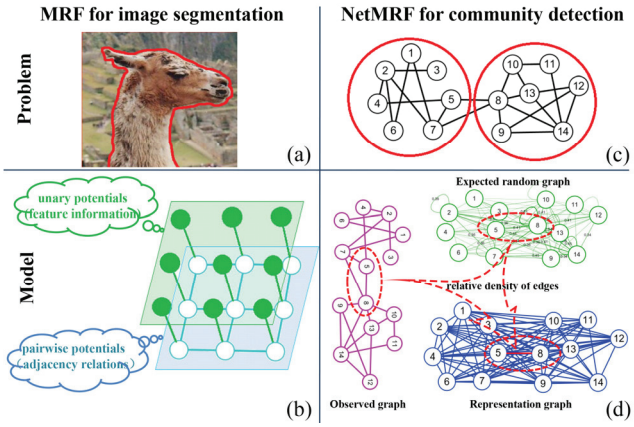


Figure 1: Comparison of models for image segmentation and network community detection. For expected random graph in (d), the width of an edge corresponds to its expected values.

To extract structural information of communities hidden in a given network and build an accurate energy function for the pMRF model, we used degree-preserved random graph null model to produce a comparison graph. Random graphs generated by this random model have not only the same numbers of nodes and edges but also the same degree for each node as in the given network, which can provide a baseline on what to expect for a graph with the same graph characteristics but with little community structure.

In summary, comparing to MRF models for image segmentation, our NetMRF model for community detection has two features. First, it does not have unary potentials due to the lack of node feature information (the green part in Figure 1(b)). Second, it introduces two auxiliary graphs, i.e., a complete graph as a graphical representation of the MRF model (the blue graph in Figure 1(d)) and a random graph for comparison (the green graph in Figure 1 (d)) so as to fully utilize the adjacency relationships among nodes.

2.1.2 The Definition of the NetMRF Model

Consider an undirected network G with n nodes and m edges. The adjacency matrix of G is $A=(a_{ij})_{n\times n}$, where $a_{ij}=1$ if nodes i and j are connected, or 0, otherwise. Suppose that the nodes fall into K communities and c_i ($c_i \in \{1, \ldots, K\}$) denotes the community that node i belongs to, and $C=(c_1,c_2, \ldots,c_{n-1},c_n)$ denotes a community partition of network G. The auxiliary graphical representation of the pMRF model is denoted by D, which is a complete graph with the same number of nodes of the given network G.

As we adopt a fully connected structure for our pMRF model, the energy function of the model is the sum of pairwise potentials for all node pairs of the given network. It should possess the property that the minimum of the energy function corresponds to the best possible community partition (Nowozin and Lampert 2011).

Pairwise potentials are defined based on the relative density of edges between a pair of nodes, i.e., the difference between the density of edges in the given network and that in a random graph. The pairwise potential between nodes i and j can then be defined as:

$$\theta_{ij}(c_i,c_j;a_{ij}) = -(-1)^{\delta(c_i,c_j)}\left(\frac{d_i d_j}{2m} - a_{ij}\right) \qquad (1)$$

where d_i is the degree of node i in G, and the δ-function $\delta(c_i,c_j)$ is 1 if $c_i=c_j$, or 0 otherwise. In Eq. (1), $d_i d_j/2m$ is the expected density of the edge between nodes i and j in the random graph. In the above potentials, if the actual density of edges between two nodes in the given network is larger (smaller) than the expected density of edges in the expected random graph, the two nodes contribute a smaller (larger) value to the pairwise potentials if they belong to the same community than when they are in different communities. As the better a community partition is, the smaller the energy function becomes, the pairwise potential imposes a soft constraint to let two densely connected nodes be more likely to be in the same community than in different ones, and likewise let two loosely connected nodes to be more likely to belong to different groups.

The energy function with exclusively the pairwise potentials of our network-specific pMRF model can then be defined as:

$$E(C;A) = \sum_{i\neq j}\theta_{ij}(c_i,c_j;a_{ij}) = \sum_{i\neq j}\left[-(-1)^{\delta(c_i,c_j)}(\frac{d_i d_j}{2m} - a_{ij})\right] \quad (2)$$

In the energy function, all pairwise constraints cooperate with each other to capture and utilize community structural signals underlying network topology to derive a global coherent community result.

With the energy function for a partition C defined, we can then adopt the Gibbs distribution, $P(C|A) \propto \exp\{-\beta E(C;A)\}$, which is a function of inverse temperature β, to compute the a *posteriori* probability of partition C given network topology A. That is

$$P(C \mid A) = \frac{1}{Z(A)}\prod_{i\neq j}\exp\left\{(-1)^{\delta(c_i,c_j)}\beta(\frac{d_i d_j}{2m} - a_{ij})\right\} \quad (3)$$

where $Z(A)$ is a normalization term, which depends on the adjacency matrix A, to ensure that $P(C|A)$ is a probability distribution. It is evident from (3) that the smaller the energy function, the larger the *a posteriori* probability. Finally, the community partition C on n nodes can be estimated as a global maximum of the posteriori of the pMRF model:

$$\hat{C} = \arg\max_{C} P(C \mid A) \quad (4)$$

2.2 Model Inference by Belief Propagation

To derive the best community partition C that maximizes the a *posteriori* probability $P(C|A)$ of the MRF model, we need to consider the inference problem on the MRF model. We may adopt one of the well-developed MAP inference methods in the MRF framework to find a configuration of community memberships of all nodes, which contribute *jointly* to the largest joint probability among all configurations in Eq. (4). The optimization of this joint probability is expected to provide a better solution than optimization of *individual* per-variable marginal probabilities $P(c_i \mid A)$, which most existing statistical models attempt to optimize (Karrer and Newman 2011; Martin, Ball, and Newman 2016). In fact, these two types of optimization usually produce different results (Bishop 2006).

Belief propagation (BP) is one of the most popular inference methods for MRF models. It has a theoretical basis and has been shown to have an excellent performance on a wide variety of problems. There are two versions of BP, the sum-product version and the max-sum version. The sum-product version of BP computes marginal probabilities of the joint probability distribution of an MRF model; the max-sum version of BP finds a configuration of variables that has the largest joint probability. Here we use the max-sum version to *jointly* maximize the joint probability.

The algorithm runs in iteration to compute a series of "messages" for each edge in a graph (the auxiliary, complete graph D) of the MRF model until convergence. According to (Nowozin and Lampert 2011), the message that node i sends to node j can be calculated iteratively in terms of all the messages that node i receives from its other neighbors k of:

$$\psi_{c_i}^{i\to j} \leftarrow \sum_{k\in N(i)\setminus j}\left[\max_{c_k}\left[(-1)^{\delta(c_i,c_k)}\beta\left(\frac{d_i d_k}{2m} - a_{ik}\right) + \psi_{c_k}^{k\to i}\right]\right] \quad (5)$$

where $N(i)$ is the neighborhood of node i in graph D. Messages are normalized in every iteration by moving the minimum of the message for any community state $c_i \in \{1,\ldots,K\}$ to zero, according to (Nowozin and Lampert 2011).

When the algorithm converges, the variable max-belief $\mu_i(c_i)$ then can be computed as:

$$\mu_i(c_i) \leftarrow \sum_{k\in N(i)}\left[\max_{c_k}\left[(-1)^{\delta(c_i,c_k)}\beta\left(\frac{d_i d_k}{2m} - a_{ik}\right) + \psi_{c_k}^{k\to i}\right]\right] \quad (6)$$

Max-beliefs are scoring functions whose maximum correspond to a community configuration with the maximal joint probability.

To find the joint maximum *a posteriori* configuration, for each variable c_i, the state with the maximum max-belief is selected, i.e.,

$$\hat{c}_i = \arg\max_{c_i\in\{1,\ldots,K\}} \mu_i(c_i) \quad (7)$$

2.3 A Max-Sum Belief Propagation Method

Since our NetMRF model is a fully connected pairwise MRF, messages on all edges of the complete graph D need to be computed in every iteration for a total of $n(n\text{-}1)$ messages. The time for computing each message is $O(n)$ according to (5). Then the whole algorithm takes $O(n^3)$ time. To make it suitable for large networks, we further developed an efficient max-sum version of BP based on the principle of network-specific fast sum-product version of BP that is used to compute marginal probabilities for other network analysis model (e.g., stochastic block model) (Decelle et al. 2011). According to (Decelle et al. 2011), following the connection between BP algorithm and the replica symmetric cavity method in statistical physics, each node can send the same message to all its non-neighbors of the original network, which is the belief of node i (i.e., $\mu_i(c_i)$), as terms of subleading order can be neglected. The messages sent to non-neighbors can be replaced by an external field. 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:

$$\psi_{c_i}^{i\to j} \leftarrow \sum_{k\in N_G(i)\setminus j}\left[\max_{c_k}\left[(-1)^{\delta(c_i,c_k)}\beta\left(\frac{d_i d_k}{2m} - 1\right) + \psi_{c_k}^{k\to i}\right]\right] + \sum_{k\notin N_G(i)}\left[\max_{c_k}\left[(-1)^{\delta(c_i,c_k)}\beta\frac{d_i d_k}{2m} + \mu_k(c_k)\right]\right] \quad (8)$$

where $N_G(i)$ is the neighborhood of node i in network G.

As messages from non-neighbors can be replaced by an external field (Decelle et al. 2011), the message that node i sends to its neighbor j can be computed based on auxiliary external field:

$$\psi_{c_i}^{i\to j} \leftarrow \sum_{k\in N_G(i)\setminus j}\left[\max_{c_k}\left[(-1)^{\delta(c_i,c_k)}\beta\left(\frac{d_i d_k}{2m}-1\right)+\psi_{c_k}^{k\to i}\right]\right]+h_{c_i}(d_i) \quad (9)$$

where $N_G(i)$ is the neighborhood of node i on network G, and $h_{ci}(d_i)$ is the external field

$$h_{c_i}(d_i)=\sum_{k=1}^{n}\left[\max_{c_k}\left[(-1)^{\delta(c_i,c_k)}\beta\frac{d_i d_k}{2m}+\mu_k(c_k)\right]\right] \quad (10)$$

The external field is a function of degree values of nodes, and there are L external fields need to be computed, where L is the number of distinct degrees.

The max-belief $\mu_i(c_i)$ can be computed based on the auxiliary external field as follow:

$$\mu_i(c_i) \leftarrow \sum_{k\in N_G(i)}\left[\max_{c_k}\left[(-1)^{\delta(c_i,c_k)}\beta\left(\frac{d_i d_k}{2m}-1\right)+\psi_{c_k}^{k\to i}\right]\right]+h_{c_i}(d_i) \quad (11)$$

In order to find a fixed point of (9) in linear time, we update the message $\psi_{c_i}^{i\to j}$, recompute $\mu_i(c_i)$, update the field $h_{c_i}(d_i)$ by adding the new contribution and subtracting the old one, and repeat until messages converge to a fixed point. The detail description of the whole efficient max-sum version of BP algorithm is as follows:

Efficient Max-Sum (*K*, *A*, *ε*, *T*)

(1) Initialize randomly K-component message vector $\psi^{i\to j}$ for each edge (i, j) on the original network G;
(2) Initialize randomly K-component max-belief μ_i for each vertex i;
(3) Compute K-component external field $h(d_i)$ according to (10) for each distanct degree value;
(4) conv $\leftarrow \varepsilon + 10$; $t \leftarrow 0$;
(5) **while** conv $> \varepsilon$ and $t < T$:
(6) **do** conv $\leftarrow 0$; $t \leftarrow t + 1$;
(7) **for** every edge (i, j) on network G:
(8) **do** Update all K-component of $\psi^{i\to j}$ according to (9);
(9) conv $\leftarrow$ conv $+ \left|\psi_{new}^{i\to j} - \psi_{old}^{i\to j}\right|$
(10) Update μ_i using the new value of $\psi^{i\to j}$ and (11);
(11) Update the field h by subtracting the old μ_i and adding the new value according to (10);
(12) Compute $\hat{c}$ from max-beliefs using (7).

For each edge (i, j), it takes $O(K^2 d_i)$ time to updating message $\psi^{i\to j}$ and max-belief μ_i according to (9) and (11) respectively, where d_i is the degree of node i, and $O(LK^2)$ time to update external field h. For a network with m edges, the time of each iteration is no greater than $O(mK^2(d_{\max}+L))$ where $d_{\max}= \max(d_1, \ldots, d_n)$. As the number of communities K is usually very small relative to the number of nodes, the time of each iteration is no greater than $O(m(d_{\max}+L))$. As the maximal number of possible iterations T is set to 100 (a constant number) in our experiments, the total time of the algorithm is $O(m(d_{\max}+L))$. As $d_{\max}$ and L are the largest node degree and the number of distinct degrees respectively, which are very small relative to the number of nodes n and can also be considered as a constant, the total time complexity of the algorithm is $O(m)$ which is nearly linear for large sparse networks.

3. Experimental Analysis

Here we validate whether our newly developed MRF approach is competitive and can outperform the existing methods, particularly statistical models.

We first tested the speedup version of our NetMFR method on two types of synthetic benchmarks, and then applied it to several widely used real-world networks with or without ground-truth of communities. We compared it with five existing methods, i.e., Karrer's method (Karrer and Newman 2011), SNMF (Wang et al. 2011), BNMTF (Zhang and Yeung 2012), MNDP (Jin et al. 2015) and DNR (Yang et al. 2016). Karrer's method is a degree-corrected stochastic block model. SNMF, BNMTF and MNDP are all nonnegative matrix factorization methods. DNR is an algorithm using deep learning. We adopted accuracy (AC) and normalized mutual information (NMI) (Liu et al. 2012) as accuracy metrics when ground-truth of communities is known, and used modularity Q (Newman and Girvan 2004) as quality metric when true communities are unknown. We compared NetMFR with its original version without speedup to assess accuracy and efficiency.

We used the programs of the existing methods from their authors and reported the results using their default parameters, expect our early deep learning method DNR (Yang et al. 2016). Besides, because all of these methods converge to local minima, we ran each method 20 times and report the result with the highest objective. For DNR, we used the results that we reported in the original paper. Since these methods all need the number of communities K, when true communities are known we set K to the ground-truth, whereas when communities are unknown we used Louvain method (Blondel et al. 2008) to find K as did in (Jin et al. 2015).

3.1 Synthetic Benchmarks

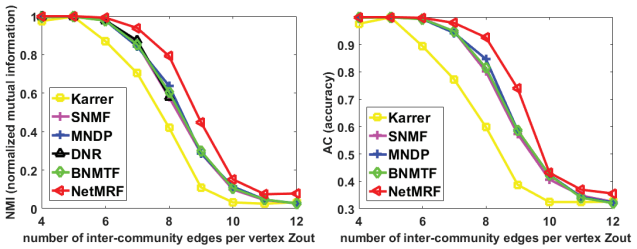
We used two types of synthetic networks, the Girvan-Newman benchmarks (Girvan and Newman 2002) and the LFR benchmarks (Lancichinetti, Fortunato, and Radicchi 2008) with known community structures.

3.1.1 The Girvan-Newman Networks

For this benchmark, each graph consists of $n = 128$ nodes divided into 4 groups of 32 nodes each. Each node has on

average z_{in} edges connecting it to members of the same group and z_{out} edges to members of other groups, with z_{in} and z_{out} chosen such that the expected degree $z_{in} + z_{out} = 16$.

Fig. 2(a) shows the accuracy of each algorithm in NMI as a function of the outside degree number z_{out} (which increases from 4 to 12 with an increment of 1). As shown, our NetMRF outperformed the other five methods. Especially when z_{out} is in the range of 7 to 9 which is a discriminative area where the superiority of NetMRF is more pronounced. We also obtained similar results when measured in AC, shown in Fig. 2(b). Collectively these results showed that our new MRF approach is more suitable for community detection, outperforming the state-of-the-arts.



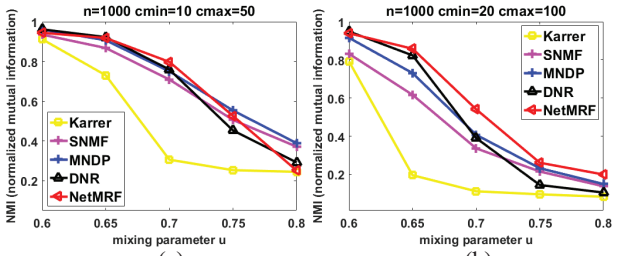
(a) (b)

Figure 2: Comparison of our NetMRF to Karrer's method, SNMF, BNMTF, MNDP and DNR in terms of (a) NMI and (b) AC on GN benchmarks. (Yang et al. 2016) reported results from z_{out}=4 to 8 in NMI, so the showed DNR results are just from 4 to 8 in (a).

3.1.2 LFR Artificial Networks

To further compare these methods, a new type of benchmark proposed by (Lancichinetti, Fortunato, and Radicchi 2008) was also used. This benchmark has some interesting properties, e.g., the heterogeneous distributions of node degree and community size that have been found in most real-world networks. As designed in (Yang et al. 2016), we considered networks with 1,000 nodes and the minimum community size c_{min} of 10 or 20. We varied the mixing parameter μ, which specifies the fraction of the links of a node connecting to nodes outside of the node's community, from 0.6 to 0.8 with an increment of 0.05, which is the most discriminative area of this benchmark. The remaining parameters were kept fixed: the average degree d was set to 20, the maximum degree d_{max} to $2.5\times d$, the maximum community size c_{max} to $5\times c_{min}$, the exponent of power-law distribution of node degrees τ_1 to -2 and community size τ_2 to -1. This design space led to two sets of benchmarks. Due to space limitation, we only present the results measured in NMI since the results in AC are also similar. We did not compare the results of BNMTF here since it cannot finish within 100 hours on every trial attempted.

Fig. 3 shows the accuracy of each algorithm in NMI as a function of the mixing parameter μ. As shown in Fig. 3(b), our NetMFR method performs the best on networks with large communities. On networks with small communities, NetMFR, SNMF, MNDP and DNR performed similarly, and were much better than Karrer's method (Fig. 3(a)). In short, our method performed competitively or better than these methods on the more challenging benchmark.



(a) (b)

Figure 3: Comparison of different methods in terms of NMI on the LFR benchmark. Each instance is averaged over 50 graphs. Shown are results on networks of (a) small communities (c_{min} = 10, c_{max} = 50) and (b) large communities (c_{min} = 20, c_{max} = 100).

3.2 Real-World Networks

Many real networks have distinct topological properties from synthetic networks on which different methods may perform differently.

3.2.1 Real Networks with Ground-Truth Communities

The comparison will be more compelling if the real networks have ground-truth of communities, so that we can use the gold metric, i.e., accuracy, to evaluate the performance of different methods.

The ten real-world networks that were analyzed (Newman 2017; Xie, Kelley, and Szymanski 2013; Sen et al. 2008) are listed in Table 1. The methods that were compared in terms of AC and NMI, along with the results on ten real networks, are listed in Tables 1 and 2. (DNR_{L2} and DNR_{CE} are two versions of DNR in (Yang et al. 2016), one is DNR with L2 norm and the other is that with cross-entropy distance.) As shown, the new method NetMRF has the best performance on 9 and 8 of the 10 networks in terms of AC and NMI, respectively. For example, using the NMI index, NetMRF is on average 10.13%, 5.45%, 6.71%, 3.88%, 7.01% and 3.58% more accurate than Karrer's method, SNMF, BNMTF, MNDP, DNR_{L2} and DNR_{CE}, respectively. We obtained similar results measured in AC. This further validates the superiority of the MRF based model over the existing models in finding communities.

Considering the efficiency, on the largest network we used, i.e., Pubmed with 19,729 nodes, our NetMFR method, Karrer's method, SNMF and MNDP ran 794s, 2,238s, 667s and 14,793s, respectively. (BNMTF could not finish in 100 hours.) We also performed the efficiency experiments on other networks though did not show them due to space limitation. The efficiency of our NetMFR is always com-

petitive with SNMF. It ran faster than the other methods compared.

Table 1: Comparison of 5 methods in terms of accuracy AC on 10 real networks with ground-truth of communities. 'Friendship6' and 'Friendship7' have the same network, but are described by different "true" communities. NDR was not included because (Yang et al. 2016) didn't reported their results in AC index.

Datasets	n	m	K	Accuracy AC (%)				
				Karrer	SNMF	BNMTF	MNDP	NetMRF
Karate club	34	78	2	97.06	**100**	**100**	**100**	**100**
Friendship6	69	220	6	81.16	78.26	60.87	78.26	**95.65**
Friendship7	69	220	7	94.20	88.41	89.86	89.86	**95.65**
Political books	105	441	3	82.86	80.95	69.52	81.90	**83.81**
US Football	115	613	12	84.35	87.83	**91.30**	**91.30**	**91.30**
Political blogs	1,490	16,717	2	87.18	94.69	94.61	94.69	**95.01**
Cora	2,708	5,429	7	37.70	42.25	40.95	44.39	**58.05**
UAI2010	3,363	45,006	19	27.78	28.52	25.51	28.92	**31.14**
Northeastern	13,882	381,935	7	**66.36**	58.87	-	56.4	65.11
PubMed	19,729	44,338	3	53.64	52.87	-	50.72	**55.53**

Table 2: Comparison of 6 methods in terms of NMI on 10 networks. We showed one decimal place for DNR's results as (Yang et al. 2016) did in their paper, and also, they did not give DNR's results on some of these networks so that we marked them as 'N/A. The '–' denotes run time >100 hours.

Datasets	NMI index (%)						
	Karrer	SNMF	BNMTF	MNDP	DNR_{L2}	DNR_{CE}	NetMRF
Karate club	83.72	**100**	**100**	**100**	**100**	**100**	**100**
Friendship6	77.02	78.64	71.22	79.30	88.8	92.4	**93.98**
Friendship7	85.10	82.11	84.30	84.26	90.7	93.2	**93.24**
Political books	54.20	56.48	51.18	53.01	55.2	58.2	**56.88**
US Football	87.06	90.38	92.42	92.42	**92.7**	91.4	92.42
Political blogs	45.68	70.95	70.78	71.07	38.9	51.7	**71.83**
Cora	17.06	24.72	26.08	33.99	N/A	N/A	**37.24**
UAI2010	20.98	23.24	21.68	25.01	N/A	N/A	**25.76**
Northeastern	**49.13**	38.66	-	40.67	N/A	N/A	45.24
PubMed	12.28	13.80	-	14.96	N/A	N/A	**16.89**

3.2.2 Real Networks without Known Communities

We often know little about the community structures of real networks. So here we further compared these methods on eight real-world networks with no known communities (Newman 2017; Nelson, McEvoy, and Schreiber 2004), which are listed in Table 3. When no "true" number of communities K is known, our method NetMRF can run on different K's (e.g., in a range of $K_{\min}$ to $K_{\max}$), and get the best K as its communities which corresponds to the smallest energy function value. However, the methods compared cannot find the number of communities automatically, which is typically suffered by most statistical model-based methods. So for fair comparison, we used Louvain method (Blondel et al. 2008) to estimate the numbers of communities and used the estimated numbers in all methods. We adopted the widely-used quality metric of modularity Q (Newman and Girvan 2004) for evaluating the quality of community structures obtained by all of these algorithms.

The results are in Table 3. As shown, NetMRF has the best performance on 7 of the 8 networks in terms of modularity Q. On average, our method is 0.2099, 0.0445, 0.0671 and 0.0404 better than Karrer's method, SNMF, BNMTF and MNDP, respectively. As Q-values are normally in the range of 0.3 to 0.8 (Newman and Girvan 2004), NetMRF obviously outperformed the existing methods.

Table 3: Comparison of 5 algorithms on 8 real networks with no known community structures. The greater a Q-value, the better. The '–' denotes run time >100 hours.

Datasets	n	m	K	Modularity Q				
				Karrer	SNMF	BNMTF	MNDP	NetMFR
Les Miserables	77	254	6	0.4575	0.5453	0.5487	0.5434	**0.5600**
Word adjacencies	112	425	7	-0.104	0.2672	0.2634	0.2712	**0.2813**
Jazz musicians	198	2,742	4	0.3696	0.4348	0.4347	0.4377	**0.4495**
C. Elegans neural	297	2,148	5	0.2617	0.3701	0.3689	0.3811	**0.4120**
E. coli metabolic	453	2,025	10	0.2656	0.3879	0.3834	0.3796	**0.4579**
E-mail network	1,133	5,451	11	0.5126	0.5007	0.4685	0.5154	**0.5712**
Power grid	4,941	6,594	39	0.1796	0.8649	0.7212	0.8683	**0.9266**
Word association	5,018	55,234	12	**0.4595**	0.3546	-	0.3613	0.4226

Table 4: Comparison of NetMRF with its original version without speedup on accuracy and runtime. "Ratio" denotes the result of NetMRF divided by that of its original version.

Datasets	NMI (%)		Ratio (%)	Time (Seconds)		Ratio (%)
	NetMRF	Original		NetMRF	Original	
Karate club	100	100	100	0.012	0.077	15.58
Friendship6	93.98	96.07	97.82	0.062	0.894	6.94
Friendship7	93.24	93.95	99.24	0.076	0.981	7.75
Political books	56.88	67.21	84.63	0.071	0.697	10.19
US Football	92.32	92.69	99.60	0.222	5.796	3.83
Political blogs	71.83	73.31	97.98	12.20	77.54	15.73
Cora	37.24	39.27	94.83	56.40	1666	3.38
UAI2010	25.76	27.62	93.27	133.7	10192	1.31
Northeastern	45.24	52.3	86.50	1962	25965	7.56
PubMed	16.89	17.69	95.48	794.0	25885	3.07

3.3 Comparison within the NetMRF Family

In order to further appreciate the effectiveness of the speedup strategy that we adopted (i.e., reducing its time from $O(n^3)$ to nearly linear) to make the MRF model practical on large networks, we compared NetMRF with its original version without speedup. As shown in Table 4, on the 10 real networks of Table 1, the NMI index of NetMRF is on average 94.94% of its original version while it ran 22.7 times faster. This also means that these two versions generate similar results but with NetMRF runs almost two orders faster than the original non-speedup version.

4. Conclusion and Discussions

For most problems of network analysis, network topology is often the only source of information available. Therefore, it is imperative to develop algorithms for network analysis, including community detection, by exclusively explore

network topological properties. In this paper, we focused on developing a new network-specific MRF approach, called NetMRF, for finding communities in large networks, which exclusively and effectively utilizes network topologies. Our extensive experiments showed that NetMRF outperforms the state-of-the-art methods on synthetic and real networks. The results suggest that NetMRF is the algorithm of choice for community detection when network topology is the only available information.

To our knowledge, NetMRF is the first approach that uses MRF for network community detection. NetMRF is general and applicable to nearly all types of real networks. NetMRF can also be readily extended to attributed network (e.g., network with content), as such content information can be directly incorporated into the unary potentials of MRF energy functions; it can also be readily extended to overlapping community detection by replacing the max-sum version of BP with the sum-product version of BP for inference in the MRF model. Therefore, the work presented here sets the foundation for a novel, general MRF based approach that is able to use network topology and content information to detect disjoint and overlapping communities.

Acknowledgments

The work was supported by Natural Science Foundation of China (61502334, 61772361, 61303110, 61373035).

References

Bernardes, D.; Diaby, M.; Fournier, R.; FogelmanSoulié, F.; and Viennet, E. 2015. A social formalism and survey for recommender systems. *ACM SIGKDD Explorations Newsletter* 16(2): 20-37.

Bishop, C. M. 2006. Pattern Recognition and Machine Learning. *Springer*.

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

Decelle, A.; Krzakala, F.; Moore, C.; and Zdeborov, L. 2011. Inference and Phase Transitions in the Detection of Modules in Sparse Networks. *Physical Review Letters* 107(6): 065701.

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

Girvan, M.; and Newman, M. E. J. 2002. Community structure in social and biological networks. *Proceedings of the National Academy of Sciences* 99(12): 7821-7826.

Jin, D.; Chen, Z.; He, D. X.; and Zhang, W. X. 2015. Modeling with node degree preservation can accurately find communities. In *Proceedings of the 29th AAAI Conference on Artificial Intelligence*, 160-167. Palo Alto, California, USA: AAAI Press.

Jin, D.; Wang, H. C.; Dang, J. W.; He D. X.; and Zhang, W. X. 2016. Detect overlapping communities via modeling and ranking node popularities. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence*, 172-178. Palo Alto, California, USA: AAAI Press.

Karrer, B.; and Newman, M. E. J. 2011. Stochastic block models and community structure in networks. *Physical Review E* 83(1): 016107.

Lancichinetti, A.; Fortunato, S.; and Radicchi, F. 2008. Benchmark graphs for testing community detection algorithms. *Physical Review E* 78(4): 046110.

Liu, H.; Wu, Z.; Li, X.; Cai, D.; and Huang, T. 2012. Constrained nonnegative matrix factorization for image representation. *IEEE Transactions on Pattern Analysis and Machine Intelligence* 34(7): 1299-1311.

Li, Y.; He, K.; Bindel, D.; and Hopcroft, J. E. 2015. Uncovering the small community structure in large networks: A local spectral approach. In *Proceedings of the 24nd International World Wide Web Conference*, 658-668. New York, NY: ACM Press.

Martin, T.; Ball B.; and Newman, M. E. J. 2016. Structural inference for uncertain networks. *Physical Review E* 93(1): 012306.

Newman, M. E. J. 2017. Real-world network data in Newman's homepage. http://www-personal.umich.edu/~mejn/netdata/.

Newman, M. E. J.; and Girvan, M. 2004. Finding and evaluating community structure in networks. *Physical Review E* 69: 026113.

Nelson, D. L.; McEvoy, C. L.; and Schreiber, T. A. 2004. The University of South Florida word association, rhyme, and word fragment norms. *Behavior Research Methods, Instruments, & Computers* 36(3): 402-407.

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

Pool, S.; Bonchi, F.; and Leeuwen, M. 2014. Description-driven community detection. *ACM Transactions on Intelligent Systems and Technology* 5: Article No.28.

Raghavan, U. N.; Albert, R.; and Kumara, S. 2007. Near linear time algorithm to detect community structures in large-scale networks. *Physical Review E* 76(3): 036106.

Rosvall, M.; and Bergstrom, C. T. 2008. Maps of random walks on complex networks reveal community structure. *Proceedings of the National Academy of Sciences* 105(4): 1118-1123.

Sen, P.; Namata, G.; Bilgic, M.; Getoor, L.; Gallagher, B.; and Eliassi-Rad, T. 2008. Collective classification in network data. *AI Magazine* 29(3): 93-106.

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

Xie, J.; Kelley, S.; and Szymanski, B. K. 2013. Overlapping community detection in networks: The state of the art and comparative study. *ACM Computing Surveys* 45: Article No.43.

Yang, L.; Cao, X. C.; He, D. X.; Wang, C.; Wang, X.; and Zhang, W. X. 2016. Modularity based community detection with deep learning. In *Proceedings of the 25th International Joint Conference on Artificial Intelligence*, 2252-2258. San Francisco, California: Morgan Kaufmann Press.

Yang, J.; and Leskovec, J. 2013. Overlapping community detection at scale: A nonnegative matrix factorization approach. In *Proceedings of the 6th ACM International Conference on Web Search and Data Mining*, 587-596. New York, NY: ACM Press.

Zhang, Y.; and Yeung, D. Y. 2012. Overlapping community detection via bounded nonnegative matrix factorization. In *Proceedings of the 18th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, 606-614. New York, NY: ACM Press.