Contents lists available at ScienceDirect

Journal of Industrial Information Integration

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

An ensemble framework for community detection

Lian Duan[*], Meral Binbasioglu

Department of Information Systems and Business Analytics, Hofstra University, USA

ARTICLE INFO

Article history:
Received 17 January 2017
Revised 25 January 2017
Accepted 27 January 2017
Available online 17 February 2017

ABSTRACT

The advances in graphs play an important role to understand interrelated data. Inside graphs, there are usually community structures where different portion of nodes are more tightly connected to form a group, and community detection has wide applications in marketing, management, health care, and education. Nowadays, many different methods are proposed to detect community structures from different perspective, but none of them can be a constant winner. Therefore, ensemble different methods can potentially improve the final result. In this paper, we present a framework for different methods to be combined for community detection, and experimental results show our framework can potentially generate a better result by different methods collectively than any single method.

1. Introduction and related work

Nowadays, with the development of data collection techniques, it is much easier for us to collect interrelated data from different perspectives. Such interrelated data can be represented as a graph data structure consisting of a set of nodes together with edges among them. These complex graphs represent systems of interactions in industrial information integration [1], biology [2], social sciences [3], healthcare [4], and business [5]. One important characteristics of graphs is that it has many different communities inside. Those communities represent different functional components in a graph. Communities in different types of graphs from different functional meanings. In industrial information integration, we have group of companies collaborating with each other for a certain product or service. In social network, we have organizations such as colleagues, friends, cities, and religious groups. In WWW, they represents different topics of webpages. Usually, nodes in these graphs are not randomly connected with each other, and they are more likely to interact with nodes from the same community. For example, industrial companies need to collaborate with each other in the same supply chain to produce products. Proteins in biology interact with each other to achieve a certain function. Searching for community structures from graphs has many meaningful applications. In industrial information integration [1], community detection will help us to narrow down the focus and find the potential partners and competitors in industrial groups. In biology, community detection can be used to find pathways of inherited diseases. In addition, it can be used for word-of-mouth marketing, customer segmentation, and influential person identification.

Most community detection methods is to find the partition that has the highest objective value through combinatorial optimization. Although the research in community detection can be traced back to 1955 where the working relationships in a government agency [6] is studied, it remains an active research topic for three major issues. First, the objective function cannot perfectly match the underlying community structure for various graphs. For example, some well-known objective functions produce intuitively wrong results for resolution limit or degeneracy [7]. Therefore, many different objective functions, such as density, modularity, and conductance, are proposed and none of them can be a constant winner. Second, the potential search space is a Bell number. This number will increase exponentially with the number of nodes in the graph. Even for a very small network with 20 nodes, its search space has $5.17E + 13$ possible combinations. With the development of information techniques, computers and electronic devices can easily collect the interaction in a real graph with millions or billions of nodes, such as Facebook, company transaction through the SWIFT international payment network, and protein interaction network. Since it is impossible to get the optimal partition by the brute-force method for network with more than hundreds of nodes, different heuristic methods, such as greedy search [8], simulated annealing [9], extremal optimization [10], and local search [11], are proposed to find a sub-optimal partition. Three, the performance evaluation of different methods are based on the comparison between the detected communities and the "latent-truth" communities. However, the so-called "latent-truth" communities for any real life graphs have errors in it. For a small-size graph, the so-called "latent-truth" communities

[*] Corresponding author.
E-mail addresses: lian.duan@hofstra.edu (L. Duan), meral.binbasioglu@hofstra.edu (M. Binbasioglu).

http://dx.doi.org/10.1016/j.jii.2017.01.001
2452-414X/© 2017 Elsevier Inc. All rights reserved.

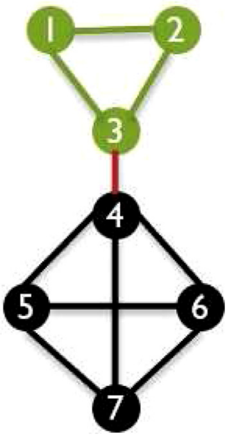

Fig. 1. A sample graph.

is usually identified by the person with a good understanding of the entire graph. For a large-size graph, the so-called “latent-truth” communities is usually identified according to different persons based on their understanding on different portions of the graph. Since the golden standard to compare is problematic, it is possible to sign a low score to a good method.

Because of various issues in community detection, we make several improvement to get better results. First, because different objective functions might be better for different graphs, we present a framework for different methods to be combined for community detection. Such combination will make our method to be more robust to different graphs. Second, we utilize a simulated graph procedure for evaluation purposes as the latent-truth in simulated graphs are genuine latent-truth.

2. Our method

In this paper, we proposed a framework for different methods to be combined for community detection. Before introducing our framework, we first introduce the two objective functions that have been selected for combination.

Given a graph G with n nodes and m edges, if the current partition P is $\{C_1, C_2, \dots, C_p\}$, we can calculate the modularity [8] for each community C_l as follows. A sample graph shown in Fig. 1 is used to understand how the modularity is calculated. The sample graph has 7 nodes and 10 edges, and the current partition has two groups: $C_1 = \{1, 2, 3\}$ and $C_2 = \{4, 5, 6, 7\}$. If an edge is randomly selected from the graph G, the probability for the selected edge with both ends in C_l is $\frac{\sum_{v_i\in C_l} k^{internal}_{v_i}}{2*m}$ where $k^{internal}_{v_i}$ is the number of internal edges for the node v_i. Take C_1 in the sample graph for example. It has three nodes: 1, 2, and 3. Both node 1 and node 2 have two internal edges and no external edges, while node 3 has two internal edges and one external edge. Therefore, $k^{internal}_{v_1} + k^{internal}_{v_2} + k^{internal}_{v_3} = 6$. In this formula, each internal edge in C_1 is counted twice. Therefore, $6/(2*10) = 0.3$ is the probability of a randomly selected edge to be the internal edge in C_1. Similarly, if an edge is randomly selected from the graph G, the probability for the selected edge with at least one end in C_l is $\frac{\sum_{v_i\in C_l} k_{v_i}}{2*m}$ where k_{v_i} is the number of edges for the node v_i. For C_1 in the sample graph, the related probability is $(2+2+3)/(2*10) = 0.35$.If an edge with one end in C_l has nothing to do with the other end in C_l, the expected probability for a randomly selected edge with both ends in C_l is $\frac{\sum_{v_i\in C_l} k_{v_i}}{2*m} * \frac{\sum_{v_j\in C_l} k_{v_j}}{2*m}$. For C_1 in the sample graph, the related probability is $0.35 * 0.35 = 0.1225$ If the nodes in community C_l are randomly selected, we are expecting that $\frac{\sum_{v_i\in C_l} k^{internal}_{v_i}}{2*m}$ will be very close to $\frac{\sum_{v_i\in C_l} k_{v_i}}{2*m} * \frac{\sum_{v_j\in C_l} k_{v_j}}{2*m}$. If two numbers are very different from each other, it is problematic for the assumption that a randomly selected edge with one end in C_l has nothing to do with the other end in C_l. In our example, the actual probability is 0.3 and the expected probability is 0.1225. Because these two numbers are very different, we believe the group C_1 is not randomly selected and likely to be a community in the sample graph. For a modularity based method [8], it tries to find the community with the highest difference between $\frac{\sum_{v_i\in C_l} k^{internal}_{v_i}}{2*m}$ and $\frac{\sum_{v_i\in C_l} k_{v_i}}{2*m} * \frac{\sum_{v_j\in C_l} k_{v_j}}{2*m}$.

Another method we select is a likelihood ratio based method [12]. In the community C_l, the total number of internal edges is $\frac{\sum_{v_i\in C_l} k^{internal}_{v_i}}{2}$. If the probability of a randomly selected edge inside the community C_l is p_{C_l}, then likelihood for us to get a graph with $\frac{\sum_{v_i\in C_l} k^{internal}_{v_i}}{2}$ edges inside the community C_l follow the binomial distribution: $Pr(C_l) = B(\frac{\sum_{v_i\in C_l} k^{internal}_{v_i}}{2}; m, p_{C_l})$. If we assume an edge with one end in C_l has nothing to do with the other end in C_l, then $p_{C_l} = \frac{\sum_{v_i\in C_l} k_{v_i}}{2*m} * \frac{\sum_{v_j\in C_l} k_{v_j}}{2*m}$ and the likelihood of getting our graph under the assumption is $Pr(C_l)_{null} = B(\frac{\sum_{v_i\in C_l} k^{internal}_{v_i}}{2}; m, \frac{\sum_{v_i\in C_l} k_{v_i}}{2*m} * \frac{\sum_{v_j\in C_l} k_{v_j}}{2*m})$. In reality, the observed chance for a randomly selected edge inside the community C_l is $\frac{\sum_{v_i\in C_l} k^{internal}_{v_i}}{2*m}$. Therefore, the likelihood of getting our graph based on the unbiased estimator is $Pr(C_l)_{observed} = B(\frac{\sum_{v_i\in C_l} k^{internal}_{v_i}}{2}; m, \frac{\sum_{v_i\in C_l} k^{internal}_{v_i}}{2*m})$. Finally, the likelihood ratio between the estimation from observed data and the estimation from null hypothesis is $\frac{Pr(C_l)_{observed}}{Pr(C_l)_{null}}$.

With the two methods we selected, we test several different ensemble methods. In order to make our study focused, we rule out the impact of different heuristic search procedure, and select greedy search, the simplest search method. In the beginning, each node form its own community, and then the most promising pair will be iteratively merged in each round. Because different objective functions will favor different pairs, an ensemble method will be promising when different methods can reveal the latent truth from different perspectives. In addition, the original value of different objective functions will change in different scales. Therefore, we ensemble the ranking position of each method instead of their original values. We tested three different ensemble function for the ranking list: product, sum, and min. For example, if the ranking of the potential merge MR_i by the method MT_j is $R_{i,j}$. If the ensemble method is product, the value of the potential merge MR_i is $\prod_{j=1}^{n} R_{i,j}$.

Fig. 2 shows how the ranking in different ensemble methods impact the choice of merge in each round through their hyperplane. Take Fig. 2(a) for example. The x-axis is the rank in the first method, and the y-axis is the rank in the second method. Any point in the same curve has the same product value. The product ensemble method will not favor one point over the other point if both are in the same curve. According to the shape of the curve, one point is more likely to be favored if its ranking in either method is close to 1. In other words, the product ensemble method will favor the merge as long as its ranking in one method is close to 1. The ranking in the other method only have marginal impact. For the sum ensemble method, the ranking in either method has equal impacts. Therefore, this method will highlight the merge that has low ranking values in both methods. For the min ensemble method, only the lower ranking value between both methods is relevant, and the other higher ranking value has no impact. As discussed in the previous paragraph, different objective functions will favor different pairs and reveal the latent truth from different