2016 IEEE 16th International Conference on Data Mining

Efficient and Scalable Detection of Overlapping Communities in Big Networks

Tianshu Lyu*, Lidong Bing†, Zhao Zhang*, Yan Zhang*
*Department of Machine Intelligence, Peking University, Beijing, China
{lyutianshu,1501214514,zhyzhy001}@pku.edu.cn
†Tencent Inc., Shenzhen, China
lbing@cs.cmu.edu

***Abstract*—Community detection is a hot topic for researchers in the fields including graph theory, social networks and biological networks. Generally speaking, a community refers to a group of densely linked nodes in the network. Nodes usually have more than one community label, indicating their multiple roles or functions in the network. Unfortunately, existing solutions aiming at overlapping-community-detection are not capable of scaling to large-scale networks with millions of nodes and edges.**

In this paper, we propose a fast overlapping-community-detection algorithm — FOX. In the experiment on a network with 3.9 millions nodes and 20 millions edges, the detection finishes in 14 minutes and provides the most qualified results. The second fastest algorithm, however, takes ten times longer to run. As for another network with 22 millions nodes and 127 millions edges, our algorithm is the only one that can provide an overlapping community detection result and it only takes 238 minutes. Our algorithm draws lessons from potential games, a concept in game theory. We measure the closeness of a node to a community by counting the number of triangles formed by the node and two other nodes form the community. Potential games ensure that the algorithm can reach convergence. We also extend the exploitation of triangle to open-triangle, which enlarges the scale of the detected communities.

***Keywords*-Community detection; Potential Games; Heuristic**

I. Introduction

Community detection is a fundamental and important work. Communities are groups of densely connected nodes in the network. Birds of a feather flock together and nodes in the same group may have certain characteristics in common in different fields, while those characteristics sometimes are not apparent to the researchers. Community detection gives the researchers a chance to gain insight into the related field. Overlapping communities allow nodes to belong to more than one community. In a social network, it is well-understood that people are naturally characterized by multiple community memberships, for instance, family circles and workmate circles.

However, no existing algorithm gives a fast resolution for overlapping community detection on large-scale networks. When the numbers of nodes and edges add up to several millions, the algorithm [1]–[5] either can only detect a rather small number of communities, or has to spend over a day giving a seemingly satisfactory result as our experiments show.

Another problem is that most community detection algorithms have no relation to a systematic theory of the emergence of communities [4]. Algorithms based on modularity optimization or dense subgraph detection exploit the structure of the network and aim to maximize the value of a global or local function (e.g. modularity [6], conductance [7]). They try to define the patterns of community structure, but the fact is that there is no acknowledged definition of community.

In this paper, we explore the problem of overlapping community detection on big networks from the perspective of community formation mechanisms. As we all know, the emergence of community is the consequence of humans' interactions. People have conflict and cooperation and tend to be with the best match friends at any time. Each player decides which community to join independently, but instead, the choice is determined by other players' choices. In mathematics, this dynamic process can be modeled by *Potential Games* in game theory. Furthermore, we find that there is a connection between *Potential Games* and heuristic algorithms. On the basis of *Potential Games*, we propose FOX (Fast Overlapping Community Search) framework, which is a principle, neat and adequate solution for community detection task. FOX is capable to process both weighted and unweighted graphs. It can also help improve the detection results provided by other algorithms.

We conclude three main contributions of our research:

- We explore the connection between *Potential Games* and heuristic algorithm and confirm that heuristic algorithm is an adequate solution for overlapping-community-detection task.
- We develop a heuristic function and its corresponding approximation on the basis of existing works. The approximation can guarantee the efficiency and quality of the detection results. Our algorithm is the fastest overlapping-community-detection algorithm to the best of our knowledge.
- Our algorithm can be used on both unweighted and weighted big graph for community detection task. Moreover, it can also be used to improve other algorithms' detection results.

II. Related Work and Background

A. Overlapping Community Detection

Community detection is a growing field of interest in many areas. Most researches focus on uncovering disjoint

2374-8486/16 $31.00 © 2016 IEEE
DOI 10.1109/ICDM.2016.17

communities. And many disjoint community detection algorithms are now available for large networks [8], [9]. But for overlapping-community detection, more common in real networks, the scalability of which is unsatisfactory. Most overlapping-community-detection algorithms are based on finding a pre-defined community structure or maximizing a mathematical criterion. The problem is that both of these two ways can't reveal the process of community emergence. And the algorithm quality and efficiency mainly rely on how to define the community structure or the objective function. Clique Percolation [10] treats adjacent cliques as communities. Generative models include Stochastic Block Model [2], [11], [12] and Nonnegative Matrix Factorization (NMF) [1]. The major limitation of NMF is the high cost of time and memory due to the matrix multiplication. Algorithms based on expanding the community locally from the seed [13]–[15]) use a benefit function (modularity, conductance and etc.) to decide which node will be absorbed into the communities. They have a good performance in scalability, while on large-scale network, the partition quality is disappointing.

Community is the product of human social activities. Individuals tend to be with people with similar interests. Everyone independently choose the best fit groups of people to join. The community emergence process can be seen as a Game as described in Section I. Game theory is firstly associated with the formation of community in [16]. Instead of optimizing a mathematical criterion, the game-theory-based algorithms are more natural and have relatively stable performance in different kinds of networks [17]. But these algorithms are not doing well on scalability [4], [18], [19].

B. Community Scoring Functions

As there is no standard definition of community, researchers have different scoring functions to assess communities. In [7], results show that Triangle Participation Ratio (TPR) performs best in density, cohesiveness and clustering coefficient and suits for heavily overlapped community.

In our algorithm, we use *WCC* (Weighted Community Clustering) [20] instead of TPR. Actually, *WCC* is put forward on the basis of TPR, not only taking the number of triangles in one community into consideration, but also counting the number of nodes that can compose a triangle. *WCC* is used in many community detection algorithms [21]–[23]. Among these algorithms, SCD [22] is a disjoint community detection algorithm and shows great advantages on large-scale networks.

III. Potential Game based Self-adaptation

We represent a social network by a weighted graph: we consider nodes to be individuals and edges to be the positive relationships (e.g. friendship, sharing the same hobby). The community formation procedure fits very naturally into the game-theoretic framework. Game players correspond to the nodes and tend to join the best fit group, which means players will always choose the strategy with the best payoff. Note that the payoff to each player depends on the strategies chosen by all players. Just like what we do in daily life, all nodes in our model constantly update their community membership according to the best responses (*best-response dynamics*). A Nash equilibrium is a list of strategies, one for each player, so that each player's strategy is a best response to all the others. Therefore, the Nash equilibrium corresponds to the best community memberships.

Potential Game [24] is a special model in game theory, in which it has been proved that the *best-response dynamics* always converges to a Nash equilibrium when the payoff for each player is related to a **global payoff function** [25]. To be more specific for our model, every node's judgement depends on whether its movement will contribute to the closeness from a global view.

Best-response dynamics has the same intuition as heuristics does. The point is, if and only if the heuristic rule is related to improve the whole partition, a heuristic algorithm is adequate to solve the community formation problem.

A. Heuristic Function: WCC

WCC [20] is a metric about the closeness between a node and a community. Given a graph $G(V, E)$, a node x and a community C,

$$WCC(x,C) = \begin{cases} \frac{t(x,C)}{t(x,V)} \cdot \frac{vt(x,V)}{|C-\{x\}|+vt(x,V-C)}, & \text{if } \ t(x,V) > 0; \\ 0, & \text{if } \ t(x,V) = 0. \end{cases} \tag{1}$$

where $t(x, C)$ stands for the number of triangles that node x closes with the nodes in C and $vt(x, C)$ stands for the number of nodes in C that close at least one triangle with node x and another node in V. C-$\{x\}$ stands for the remaining part of C when taking out x. Together, the level of closeness between node x and community C is denoted by $WCC(x, C)$.

We expand this metric further for the sake of overlapping-community detection. The WCC value of a community C_i is the sum of its members' WCC value.

$$WCC(C_i) = \sum\nolimits_{x \in C_i} WCC(x, C_i) \tag{2}$$

For a community partition P=$\{C_1, C_2, \ldots, C_k\}$,

$$WCC(P) = \sum\nolimits_{i=1}^{k} WCC(C_i) \tag{3}$$

One node in P changes its community membership and the new partition is P'. We define the heuristic function Δ as $WCC(P')$-$WCC(P)$.

We further develop the concept of *open-triangle*, which is a structure composed of 3 nodes and 2 edges. Triangle describes the condition that three people are mutual friends, while *open-triangle* is the phenomenon that two strangers have one common friend. In the following parts, s-FOX denote as the algorithm using *open-triangle*.

B. Self-adaptation Strategies

During every iteration, nodes have 4 strategies of movements: (1)do not move, (2)leave the community and be alone, (3)transfer to another community and (4)stay and at the same time join in another community. The last choice makes the community overlapped. The benefit of every strategy is estimated by the heuristic function, and every node tends to make the best choice to maximize the heuristic function Δ. Next we will derive the payoff of the 4 strategies (denoted as Δ_S, Δ_L, Δ_T and Δ_C respectively).

[Strategy 1] Stay and do not move: The partition doesn't change at all.

$$\Delta_S = WCC(P) - WCC(P) = 0$$

[Strategy 2] Leave and be alone: Suppose the original partition is P=$\{C_1, C_2, \ldots, C_k\}$ and when node x leaves its community C_k, the partition P'=$\{C_1, C_2, \ldots, C_k', \{x\}\}$, where C_k=$C_k' \bigcup \{x\}$

$$\Delta_L(x, C_k) = WCC(P') - WCC(P)$$

Especially when community C_k is a singleton community, $\Delta_L(x, C_k) = 0$.

[Strategy 3] Transfer to another community: Suppose that node x transfers from C_1 to C_k, the original partition is P=$\{C_1, C_2, \ldots, C_k\}$ and the new partition is P=$\{C_1', C_2, \ldots, C_k'\}$, where C_1=$C_1' \bigcup \{x\}$ and C_k'=$C_k \bigcup \{x\}$. This movement is actually a composite transformation of 2 steps. Step 1: node x leaves C_1 and doesn't join any community, P_m=$\{C_1', C_2, \ldots, C_k, \{x\}\}$. And step 2: node x join C_k, P'=$\{C_1', C_2, \ldots, C_k'\}$. We can easily figure out that step 2 is an inverse transformation of Strategy 2.

$$\begin{aligned}\Delta_T &= (WCC(P') - WCC(P_m)) + (WCC(P_m) - WCC(P)) \\ &= \Delta_L(x, C_1) - \Delta_L(x, C_k)\end{aligned}$$

We denote the best transferred community as the community which has the biggest $\Delta_L(x, C_k)$. The WCC improvement of this best transferring choice is $\Delta_T(x, C_{best})$

[Strategy 4] Do not move and at the same time join in another community: Suppose that node x copies itself to C_k, P=$\{C_1, C_2, \ldots, C_k\}$ and P'=$\{C_1, C_2, \ldots, C_k'\}$, where C_k'=$C_k \bigcup \{x\}$. Also, this is a composite transformation. The intermediate state is P_m=$\{C_1, C_2, \ldots, C_k, \{x\}\}$. Similarly, the WCC improvement of the best community is $\Delta_C(x, C_{best})$.

$$\begin{aligned}\Delta_C &= (WCC(P') - WCC(P_m)) + (WCC(P_m) - WCC(P)) \\ &= -\Delta_L(x, C_k') + WCC(x, \{x\}) = -\Delta_L(x, C_k')\end{aligned}$$

For all of these 4 strategies, x will choose the one that can maximize the payoff.

$$Strategy(x) = max(\Delta_S(x), \Delta_L(x), \Delta_T(x, C_{best}), \Delta_C(x, C_{best}))$$

Simply stated, in every iteration, if x is negative for its community, it will be removed from the community anyway. If x also hurts all of the other communities, it will be alone. Otherwise x will transfer to the best suitable community. But when x is beneficial to its community, it will stay there and consider whether to join other communities which it can bring the most benefit. Here, the benefit is the WCC improvement namely the enhancement of connectedness.

IV. Algorithm

FOX and s-FOX are aiming at detecting communities of unweighted or weighted communities by counting triangles and open-triangles respectively. For simplicity, the following discussion is based on unweighted graph.

A. Pre-treatment

To initialize the first partition, we employ *local clustering coefficient*, which is well-matched to the main procedure of FOX.

We first compute the CC of all nodes and rank them in decreasing order. Next, for the top-ranked node, all of its neighbors are marked as visited and added into a community. Then in the unvisited node set, pick the top-ranked node and its unvisited neighbors out to form a community. This work will stop when all nodes have been visited.

B. Best response dynamics

Best response dynamics is the community formation progress. From the analysis above, we can find that the computation of $\Delta_L(x, C)$ plays a significant role. Both Δ_T and Δ_C are computed on the basis of Δ_L. In order to apply our algorithm to large-scale network, we propose an approximation from a statistical standpoint and the complexity decreases notably to $O(n)$.

1) ***Counting triangles approximately:*** We assume that the more edges between x and C, the more triangles between x and C. Based on this assumption, we approximately calculate the number of triangles by further assuming that: (1) in a community, any two nodes are connected with the same probability; (2) every edge closes at least one triangle in densely overlapped network; (3) all nodes' local clustering coefficients have similar values. When node x is outside the community C, we approximately calculate the number of triangles as

$$\hat{t}(x, C) = \binom{d_C}{2} \cdot p \tag{4}$$

$$\hat{t}(x, V - C) = \binom{d_{V-C}}{2} \cdot cc \tag{5}$$

$$\hat{vt}(x, V - C) = d_{V-C} \tag{6}$$

where V is the node set of the graph, p is the probability that two random nodes in community C are connected, d_C is the number of edge between C and x and the clustering coefficient of the graph is cc. When node x is a member of C, the approximation is similar.

The approximation calculation of open-triangle is similar to the calculation of triangles. Suppose that node x is outside the community C and nodes y and z belong to C, there are two types of open-triangles that x y z can form: $z - x - y$, and $x - y - z$.

$$\hat{t_s}(x, C) = \binom{d_C}{2} + d_C \cdot (|C| - d_C) \cdot p \tag{7}$$

*2) **Classifying the nodes in C'_k into 2 types:*** Continuing the analysis of **Strategy 2**, the difference between P and P' is the departure of node x. Only node x and the nodes in community C_k get a new WCC after this movement. Therefore, when calculating the difference of $WCC(P)$, we only have to calculate the WCC change of nodes in C_k including x.

$$\begin{aligned}\Delta_L(x, C_k) = &\textstyle\sum_{n \in C'_k} (WCC(n, C'_k) - WCC(n, C'_k \bigcup \{x\})) \\ &- WCC(x, C'_k \bigcup \{x\})\end{aligned} \tag{8}$$

For all the nodes in C'_k, they can be divided into 2 categories, node sets N and M. Nodes in N are the neighbors of node x, and nodes in M are not. Assume that nodes and edges density in every segment of the whole graph are homogeneous. Segments include N, M and G-C'_k. Next, we can calculate the WCC of nodes in N and M respectively to further simplify Equation 8.

$$\begin{aligned}\Delta_L(x, C_k) = &\textstyle\sum_{n \in N} (WCC(n, C'_k) - WCC(n, C'_k \bigcup \{x\})) \\ &+ \textstyle\sum_{n \in M} (WCC(n, C'_k) - WCC(n, C'_k \bigcup \{x\})) \\ &- WCC(x, C'_k \bigcup \{x\}) \\ &= |N| \cdot \Delta(a) + |M| \cdot \Delta(b) - WCC(x, C'_k \bigcup \{x\})\end{aligned}$$

$\Delta(a)$ denotes the average WCC difference of the nodes in N when x leaves from C_k. Or a can just be seen as a random node in N. $\Delta(b)$ is the average difference of nodes in M.

$$\Delta(n) = WCC(n, C'_k) - WCC(n, C'_k \bigcup \{x\})$$

Next, we discuss these 3 kinds of nodes, nodes in N, nodes in M and node x, respectively. The statistics we need are

- d_{in}: the number of edges between x and C_k
- d_{out}: the number of edges between x and G-C_k
- p_{in}: the probability that two nodes in C_k are connected by an edge
- p_{ext}: the clustering coefficient of the graph
- q: the average number of edges between nodes in C_k and nodes in G-C'_k-$\{x\}$
- S: the size of C'_k
- p: the average degree of the whole graph

When one node finishes implementing its best strategy, these statistics also need to be updated. The computation complexity is $O(d)$, where d is the average degree.

With the help of these statistics and Equation 4 5 6, we can approximately calculate Equation 1 and the value of $\Delta_L(x, C)$. The specific derivation is in Appendix B.

$$\Delta(a) = \frac{(d_{in}-1)p_{in}}{0.5(S-1)(S-2)p_{in}^3 + (d_{in}-1)p_{in} + q(S-1)p_{in}p_{ext} + 0.5S(S-1)p_{ext} + d_{out}p_{ext}} \frac{(S-1)p_{in}+1+q}{S+q}$$

$$\Delta(b) = -\frac{0.5(S-1)(S-2)p_{in}^3}{0.5(S-1)(S-2)p_{in}^3 + q(q-1)p_{ext} + q(S-1)p_{in}p_{ext}} \frac{(S-1)p_{in}+q}{(S+q)(S-1+q)}$$

$$WCC(x, C'_k \bigcup \{x\}) = -\frac{(d_{in}(d_{in}-1)p_{in})(d_{in}+d_{out})}{(d_{in}(d_{in}-1)p_{in}) + d_{out}(d_{out}-1)p_{ext}} \frac{1}{S+d_{out}}$$

C. Post-treatment

Nodes move in turns and this will inevitably bring a problem of community connectivity. We are curious that this problem isn't mentioned in [4], [19], [22]. We put forward two strategies coping with the problem. First is the order of nodes. Nodes with higher degree move first, as they are more influential . Second, we conduct a connectivity analysis after the algorithm all the unconnected communities will be marked. All the nodes in unconnected communities will then choose the best community, which must be connected, to join into. Some small communities may be entirely included in some bigger communities. This kind of small community is ignored in the final partition result.

D. Is the Approximation Reasonable?

To prove that our approximation is reasonable, we also develop an algorithm, FOX-naive, in which $\Delta_L(x, C_k)$ is precisely calculated. The termination criteria of FOX-naive and FOX are also different. In FOX-naive, best response dynamics is perfectly performed, and at last all nodes satisfy with their situations and choose to stay in their communities. But in FOX, the approximation is not precise enough to reach the stable condition as FOX-naive does. In FOX, after each iteration, we compute the exact value of potential function $WCC(P)$. When the difference between the $WCC(P)$ of two iterations is less than a threshold t, the algorithm stops. In Fig. 1 and Fig. 2, the experiment on three small datasets, we find that there are no obvious differences between the results provided by FOX-naive and FOX.

V. Experiments

A. Experimental set-up

We test our algorithm on two types of datasets: networks with ground-truth and real large-scale networks, as given in Table I. The configuration of our computer is: two Intel(R) Xeon(R) CPU E5-2620 at 2.00GHz, 64GB of RAM. In FOX and s-FOX, the threshold t is set to 0.1%.

*1) **Datasets:*** **Network with ground-truth.** Most overlapping-detection algorithms use the benchmark datasets provided by SNAP [26]. We employ the collaboration network of DBLP, Amazon product co-purchasing network, and Youtube social network.

Real large-scale networks. We analyze 2 big networks: a mobile communication network and the Google+ mutual followed relation network.

The mobile communication network is a phone-call record of a city, including 3.9 million users and lasting for 3 months. We draw an edge between 2 nodes only if the two users call each other more than one time. Google+ network is much bigger. The authors of [27] collected a weakly connected component of Google+, which includes over 70% of all Google+ users from Jul. 2011 to Oct. 2011. We only reserve the mutual followed relationships in the raw data.

TABLE I

BASIC INFORMATION OF OUR DATASETS. N: NUMBER OF NODES. E: NUMBER OF EDGES. D: AVERAGE DEGREE. D_{max}: MAXIMUM DEGREE. C: MAXIMAL CONNECTED GRAPH COVERAGE. N_c: NUMBER OF COMMUNITIES. CC: CLUSTERING COEFFICIENT. M: MILLION. K: THOUSAND.

Data Sets	N	E	D	D_{max}	C	N_c	CC	Node	Edge	Community
DBLP	317K	1M	6.62	343	100%	13K	0.63	author	co-author	publication venue
Amazon	335K	926K	5.53	549	100%	75K	0.40	product	co-purchased	products category
Youtube	1.1M	3.0M	5.27	28754	100%	8K	0.08	user	follow	interest group
Phone-call	3.9M	20.5M	10.25	438	94%	-	0.12	user	make phone call	-
Google+	22.5M	127.3M	9.98	7347	94%	-	0.24	user	follow	-

2) ***Baseline:*** Compared baseline algorithms including disjoint-community-detection algorithms (Louvain [13], Infomap [3], SCD [22]) and overlapping-community-detection algorithms (SVI [2], GAME [4], BigCLAM [1], OSLOM [14], FOX-naive in Sec. IV-D).

3) ***Metrics:*** As for the datasets with ground-truth, the main goal of the experiment is to evaluate the similarity between the ground-truth and the detected result. We use two evaluation metrics: Average **F1-Score** (F1) [1] and **Normalized Mutual Information** (NMI) [5]. Both of these two values are in [0,1], with 1 standing for perfect matching.

The phone-call record network and Google+ network have no ground-truth communities. We evaluate the quality of the detected communities with the following metrics.

Density is the average probability of the nodes in the same community being connected.

$\boldsymbol{w_c/w_i}$ is proposed to qualify community partition of phone-call record network [28].We define the edge weight w as the cumulative time of the phone calls between 2 users. w_c denotes the average edge weight of the edges inside the community, and w_i denotes the average weight of inter-community edges of the community. The higher the ratio of w_c/w_i is, the more intensive the communication within a community is, compared with the inter-community communication.

Modularity is adapted to measuring overlapping community by [29].

$$Q_{ov} = \frac{1}{2m}\sum\nolimits_{c \in C}\sum\nolimits_{i,j \in V}[r_{ijc}A_{ij} - \frac{s_{ic}k_i s_{jc}k_j}{2m}]$$

Please refer to [29] for the detail information about this metric.

B. Experiments using ground-truth

1) ***Comparing with overlapping-community-detection algorithm:*** Fig. 1 and Fig. 2 present the performance of our two algorithms (i.e. FOX and s-FOX), FOX-naive and three baseline algorithms on three networks with ground-truth. Game run for over 3 days on Amazon dataset and achieved less than 5% of the total progress, so we terminated it and do not include its results in these two figures. FOX and FOX-naive perform very similar on these three datasets under NMI and F1 score. But, take the experiment on the Youtube dataset as an example, FOX-naive took 3 hours to finish the detection, while FOX only took 8 mins. It shows that FOX achieves encouraging efficiency improvement without sacrificing the effectiveness of community detection, which demonstrates the reasonability of our approximation in FOX. s-FOX and OSLOM get close scores, following FOX. Communities detected by s-FOX is bigger than that by FOX, as it contains more periphery nodes. So the results are not as good as FOX does. We have no acknowledged definitions of communities and the boundaries of communities are actually blurred. SVI and BigCLAM require the number of communities as input. We tend to set it as the number of ground-truth communities. But it's too large for SVI to finish the detection task. Therefore, we set the community number to 1000 for SVI to finish the detection.

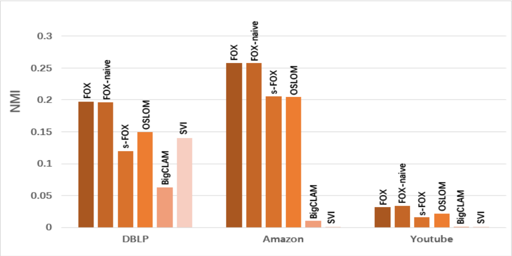

Fig. 1. NMI with ground-truth

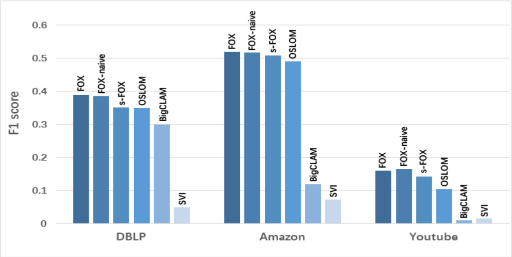

Fig. 2. F1-score with ground-truth

C. Experiments on big networks

There are a variety of problems for the compared methods to run on big networks, so that we have to make some compromise. SVI cannot handle the phone-call dataset because of its huge memory consumption. Game, as stated before, takes too much time to finish. BigCLAM can decide how many communities to detect automatically. But in this mode, BigCLAM cannot provide a result in three days. We set several community numbers for testing, and 150,000 is the biggest one that BigCLAM can finish detecting with an acceptable running time (38 hours).

TABLE II
EVALUATION RESULTS ON PHONE-CALL AND GOOGLE+ DATASETS

Dataset	phone-call				Google+		
Algorithm	time cost	$Density$	w_c/w_i	Q_{ov}	time cost	$Density$	Q_{ov}
BigCLAM	38 hr.	0.028	0.604	**1.401**	-	-	-
OSLOM	194 min	0.442	1.845	0.621	-	-	-
FOX	**14 min**	**0.607**	2.920	0.758	238 min	0.529	1.044
s-FOX	20 min	0.325	**19.434**	0.936	260 min	0.328	1.334

Table II shows the algorithm performances on two datasets. FOX is the fastest overlapping-community-detection algorithm. Although FOX and s-FOX do not take edge weight into account, their detection results are still better than BigCLAM and OSLOM on w_c/w_i. In the communities provided by OSLOM, the percentage of triangle sums up to 13.1%. However, in the communities provided by s-FOX the percentage is 2.5%. Triangle contributes a lot to edge density, as the edge density of triangle always equals 1. BigCLAM performs brilliantly in Q_{ov} and ours is the second best. But BigCLAM performs poorly in the other aspects.

The volume of Google+ data is about six times more than the volume of phone-call record. Our algorithms finish the detection work for about 4 hours, which is the only overlapping-community-detection algorithm that can accomplish this work. OSLOM aborted when an exception occurred after running over 3 days. As shown in Table I, the Google+ network has the similar average degree as the phone-call network has, which shows Google+ is a quite dense network with large number of nodes, thus it is a very challenging dataset for community detection algorithms. As shown in Table II, our algorithms' the run time for Google+ is about fifteen times more than that for the phone-call data.

VI. Conclusion

In this paper, a fast overlapping-community-detection algorithm is proposed. When the scale of data is over 10 millions, it can provide a reasonable community partition within hours, which is the best performance on both accuracy and time-cost. This heuristic algorithm learns from potential games in game theory. Moreover, the approximation of the heuristic function ensures the quality of community and the speed of the detection. Our algorithm is also appropriate to the weighted graph, only if the edge weight is proportional to the closeness of relationship. For more details about this work, please refer to the full version[1].

Acknowledgment

This work is supported by 973 Program with Grant No. 2014CB340405, NSFC with Grant No. 61532001 and No. 61370054, and MOE-RCOE with Grant No. 2016ZD201. We thank the anonymous reviewers for their valuable comments.

References

[1] J. Yang and J. Leskovec, "Overlapping community detection at scale: A nonnegative matrix factorization approach," in *WSDM'13*, pp. 587–596.

[2] P. K. Gopalan and D. M. Blei, "Efficient discovery of overlapping communities in massive networks," *PNAS*, vol. 110, no. 36, pp. 14 534–14 539, 2013.

[3] M. Rosvall and C. T. Bergstrom, "Maps of random walks on complex networks reveal community structure," *PNAS*, vol. 105, no. 4, pp. 1118–1123, 2008.

[4] W. Chen, Z. Liu, X. Sun, and Y. Wang, "A game-theoretic framework to identify overlapping communities in social networks," *DMKD*, vol. 21, no. 2, pp. 224–240, 2010.

[5] A. Lancichinetti, S. Fortunato, and J. Kertész, "Detecting the overlapping and hierarchical community structure in complex networks," *New Journal of Physics*, vol. 11, no. 3, p. 033015, 2009.

[6] M. E. Newman, "Modularity and community structure in networks," *PNAS*, vol. 103, no. 23, pp. 8577–8582, 2006.

[7] J. Yang and J. Leskovec, "Defining and evaluating network communities based on ground-truth," vol. 42, no. 1, 2015, pp. 181–213.

[8] R.-H. Li, L. Qin, J. X. Yu, and R. Mao, "Influential community search in large networks," *VLDB'15*, pp. 509–520.

[9] J. Shao, Z. Han, Q. Yang, and T. Zhou, "Community detection based on distance dynamics," in *KDD'15*, pp. 1075–1084.

[10] B. Adamcsek, G. Palla, I. J. Farkas, I. Derényi, and T. Vicsek, "Cfinder: locating cliques and overlapping modules in biological networks," *Bioinformatics*, vol. 22, no. 8, pp. 1021–1023, 2006.

[11] M. Zhou, "Infinite edge partition models for overlapping community detection and link prediction," in *AISTATS'15*, pp. 1135–1143.

[12] A. Anandkumar, R. Ge, D. Hsu, and S. M. Kakade, "A tensor approach to learning mixed membership community models," *JMLR*, vol. 15, no. 1, pp. 2239–2312, 2014.

[13] 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*, vol. 2008, no. 10, p. P10008, 2008.

[14] A. Lancichinetti, F. Radicchi, J. J. Ramasco, and S. Fortunato, "Finding statistically significant communities in networks," *PloS one*, vol. 6, no. 4, p. e18961, 2011.

[15] J. J. Whang, D. F. Gleich, and I. S. Dhillon, "Overlapping community detection using neighborhood-inflated seed expansion," *TKDE*, vol. 28, no. 5, pp. 1272–1284, 2016.

[16] S. Athey, E. Calvano, and S. Jha, "A theory of community formation and social hierarchy," *preprint*, 2006.

[17] J. Xie, S. Kelley, and B. K. Szymanski, "Overlapping community detection in networks: The state-of-the-art and comparative study," *ACM Computing Surveys*, vol. 45, no. 4, p. 43, 2013.

[18] L. Zhou, P. Yang, K. Lü, L. Wang, and H. Chen, "A fast approach for detecting overlapping communities in social networks based on game theory," in *Data Science*. Springer, 2015, pp. 62–73.

[19] R. Narayanam and Y. Narahari, "A game theory inspired, decentralized, local information based algorithm for community detection in social graphs," in *ICPR'12*, pp. 1072–1075.

[20] A. Prat-Pérez, D. Dominguez-Sal, J. M. Brunat, and J.-L. Larriba-Pey, "Shaping communities out of triangles," in *CIKM'12*. ACM.

[21] Y. Song, S. Bressan, and G. Dobbie, "Fast disjoint and overlapping community detection," in *Transactions on Large-Scale Data-and Knowledge-Centered Systems XVIII*. Springer, 2015, pp. 153–179.

[22] A. Prat-Pérez, D. Dominguez-Sal, and J.-L. Larriba-Pey, "High quality, scalable and parallel community detection for large real graphs," in *WWW'14*, pp. 225–236.

[23] M. Saltz, A. Prat-Pérez, and D. Dominguez-Sal, "Distributed community detection with the wcc metric," in *WWW'15*, pp. 1095–1100.

[24] R. W. Rosenthal, "A class of games possessing pure-strategy nash equilibria," *International Journal of Game Theory*, vol. 2, no. 1, pp. 65–67, 1973.

[25] N. Nisan, T. Roughgarden, E. Tardos, and V. V. Vazirani, *Algorithmic game theory*. Cambridge University Press, 2007.

[26] J. Leskovec and A. Krevl, "SNAP Datasets: Stanford large network dataset collection," http://snap.stanford.edu/data, Jun. 2014.

[27] N. Z. Gong, W. Xu, L. Huang, P. Mittal, E. Stefanov, V. Sekar, and D. Song, "Evolution of social-attribute networks: measurements, modeling, and implications using google+," in *IMC'12*, pp. 131–144.

[28] G. Palla, A.-L. Barabási, and T. Vicsek, "Quantifying social group evolution," *Nature*, vol. 446, no. 7136, pp. 664–667, 2007.

[29] V. Nicosia, G. Mangioni, V. Carchiolo, and M. Malgeri, "Extending the definition of modularity to directed graphs with overlapping communities," *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2009, no. 03, p. P03024, 2009.

[1] http://www.cis.pku.edu.cn/faculty/system/zhangyan/papers/fox_full.pdf