COMMUNITY DETECTION IN SOCIAL NETWORKS USING EXTENDED SELF ORGANIZING MAP ALGORITHM

Harish Kumar Shakya[1], Kuldeep singh[2], Bhaskar Biswas[3]

[1,2]PhD Scholar, Indian Institute of Technology (Banaras Hindu University), Varanasi, (India)

[3]Assistant Professor, Department of CSE,

Indian Institute Of Technology (Banaras Hindu University), Varanasi, (India)

ABSTRACT

Social networks are often studied as graphs, and detecting communities in a social network can be modeled as a seriously non linear optimization problem. Soft computing techniques have shown promising results for solving this problem. In this paper, we have proposed a new approach based on self organizing map to community detection. By using a proper weight updating scheme, a network can be organized into dense sub graphs according to the topological connection of each node. A community is usually defined in a qualitative way, as a subset of nodes of a network which are more densely connected among them -selves than to the rest of the networks.

Keywords: Community Detection, Modularity, Neural Network, Self Organizing Map, Social Networks

I. INTRODUCTION

Social network analysis is the mapping and measuring of relationships and flows between people, groups, organizations, computers, URLs, and other connected information/knowledge entities. The nodes in the network are the people and groups while the links show relationships or flows between the nodes. SNA provides both a visual and a mathematical analysis of human relationships. Management consultants use this methodology with their business clients and call it Organizational Network Analysis.

The process of social network analysis typically involves the use of questionnaires and/or interviews to gather information about the relationships between a defined group or network of people. The responses gathered are then mapped using a software tool specifically designed for the purpose. This data gathering and analysis process provides a baseline against which you can then plan and prioritize the appropriate changes and interventions to improve the social connections and knowledge flows within the group or network. Key stages of the process will typically include:

- Identifying the network of people to be analyzed (e.g. team, workgroup, and department).
- Gathering background information - interviewing managers and key staff to understand the specific needs and problems.
- Clarifying objectives, defining the scope of the analysis and agreeing on the level of reporting required.

- Formulating hypotheses and questions.
- Developing the survey methodology and designing the questionnaire.
- Surveying the individuals in the network to identify the relationships and knowledge flows between them.
- Use a software mapping tool to visually map out the network.
- Reviewing the map and the problems and opportunities highlighted using interviews and/or workshops.
- Designing and implementing actions to bring about desired changes.
- Mapping the network again after a suitable period of time.

Social network analysis is used extensively in a wide range of applications and disciplines. Some common network analysis applications include data aggregation and mining, network propagation modeling, network modeling and sampling, user attribute and behavior analysis, community-maintained resource support, location-based interaction analysis, social sharing and filtering, recommender systems development, and link prediction and entity resolution [1]. In the private sector, businesses use social network analysis to support activities such as customer interaction and analysis, marketing, and business intelligence needs. Some public sector uses include development of leader engagement strategies, analysis of individual and group engagement and media use, and community-based problem solving.

Social network analysis is also used in intelligence, counter-intelligence and law enforcement activities. This technique allows the analysts to map a clandestine or covert organization such as a espionage ring, an organized crime family or a street gang. The National Security Agency (NSA) uses its clandestine mass electronic surveillance programs to generate the data needed to perform this type of analysis on terrorist cells and other networks deemed relevant to national security. The NSA looks up to three nodes deep during this network analysis [3]. After the initial mapping of the social network is complete, analysis is performed to determine the structure of the network and determine, for example, the leaders within the network[3].This allows military or law enforcement assets to launch capture-or-kill decapitation attacks on the high-value targets in leadership positions to disrupt the functioning of the network. The NSA has been performing social network analysis on Call Detail Records (CDRs), also known as metadata, since shortly after the September 11 Attacks.

1.1 Community Detection

The study of networks (a set of nodes interconnected by links) has become a ubiquitous topic in many branches of science. This is because many systems of interest can be represented in this way, as for example, Internet, the WWW, food webs, neural networks, communication networks, social networks etc. Many different properties have been revealed as: small world effect, high network transitivity, power law degree distributions, etc.

Community structure is defined, in a qualitative way, as the possibility of recognizing within the networks, subsets of nodes which are more connected among themselves that to the rest of the network. If we can detect such structures, we will get information of practical importance. Such groups in the WWW might correspond to sets of web pages on related topics, in the case of social networks; they would indicate groups that share interests, problems etc. In a metabolic network, it might help to identify groups of nodes which perform different functions [4].

In the study of networks, such as computer and information networks, social networks or biological networks, a number of different characteristics have been found to occur commonly, including the small-world property, heavy-tailed degree distributions, and clustering, among others. Another common characteristic is community structure. In the context of networks, community structure refers to the occurrence of groups of nodes in a

network that are more densely connected internally than with the rest of the network. The communities are often defined in terms of the partition of the set of vertices that is each node is put into one and only one community, Not all networks need display community structure. Many model networks, for example, such as the random graph and the Barabási–Albert model, do not display community structure. In this paper, we present a new modularity-based approach where we find the number of communities:

- We input a range of tentative communities in which we feel the graph can be partitioned.
- Output is the number of communities giving the highest modularity value.

1.2 Self Organizing Map

A self-organizing map (SOM) or self-organizing feature map (SOFM) is a type of artificial neural network (ANN) that is trained using unsupervised learning to produce a low-dimensional (typically two-dimensional), discretized representation of the input space of the training samples, called a map. Self-organizing maps are different from other artificial neural networks in the sense that they use a neighborhood function to preserve the topological properties of the input space. This makes SOMs useful for visualizing low-dimensional views of high-dimensional data, akin to multidimensional scaling. The model was first described as an artificial neural network by the Finnish professor Teuvo Kohonen, and is sometimes called a Kohonen map or network.

Like most artificial neural networks, SOMs operate in two modes: training and mapping. "Training" builds the map using input examples (a competitive process, also called vector quantization), while "mapping" automatically classifies a new input vector.

A self-organizing map consists of components called nodes or neurons. Associated with each node is a weight vector of the same dimension as the input data vectors and a position in the map space. The usual arrangement of nodes is a two-dimensional regular spacing in a hexagonal or rectangular grid. The self-organizing map describes a mapping from a higher-dimensional input space to a lower-dimensional map space. The procedure for placing a vector from data space onto the map is to find the node with the closest (smallest distance metric) weight vector to the data space vector. While it is typical to consider this type of network structure as related to feed forward networks where the nodes are visualized as being attached, this type of architecture is fundamentally different in arrangement and motivation. Here we use the self organizing maps for community detection in social networks.

II. EXPERIMENTAL WORK

The aim is to learn a feature map from the spatially continuous input space, in which our input vectors live, to the low dimensional spatially discrete output space, which is formed by arranging the computational neurons into a grid. The stages of the SOM algorithm that achieves this can be summarized as follows:

- Initialization – Choose random values for the initial weight vectors w^j .
- Sampling – Draw a sample training input vector x from the input space.
- Matching – Find the winning neuron I(x) that has weight vector closest to the input vector, i.e. the minimum value of $d_j(x) = \sum (x_i - w_{ji})^2$.
- Updating – Apply the weight update equation $\Delta w_{ji} = \eta(t)\, T_{j,I(x)}(t)\, (x_i - w_{ji})$; where $T_{j,I(x)}(t)$ is a Gaussian neighborhood and $\eta(t)$ is the learning rate.
- Continuation – keep returning to step 2 until the feature map stops changing.

In this paper, we have implemented an adaptation of the basic SOM algorithm as given in [1].According to the topological connection of each node, our approach automatically organizes a network into dense sub-graphs without any heuristic manipulation. Besides unweighted undirected networks, our method can also be used to detect communities in both weighted and bipartite networks.

2.1 Basic Functions and Definition

G= the given network

n= number of nodes

m= be an upper bound of the number of communities

$A = [a_{ij}]_{nxn}$ be the adjacency matrix of the network G

B = A+I (I is a unit matrix)

The scheme of the self-organizing map for detecting community structure, has n input neurons corresponding to the nodes v1,v2, ⋯ ,vn in G, and m output neurons representing putative communitiesC1,C2, ⋯ ,Cm. For each node i, the learning input Bi ∈ R n is a vector such that $b_{i,i}$ =1 and $b_{j,i}$ = 1 if and only if v_j is adjacent to v_i in G, $b_{j,i}$ = 0, otherwise. The connection weight matrix between input neurons and output neurons is $W = [w_{ij}]_{nxm}$, where the weight w_{ij} expresses the possibility or membership degree that node vi belongs to community C_j. If the input neuron vi is mapped into the output neuron C_j, then the connection between viand C_j should be reinforced [2]. Moreover, all other weights of the winner output neuron C_j are also modified according to the adjacency relationship between the corresponding nodes and the input node v_i, since two adjacent nodes are more likely to be in a same community. The details of our implementation of the SOM algorithm are described as follows:

- Step 1. Initialization
  Set the initial learning rate a0 and the maximum number of iterations MaxIter.
  Randomly initiate$W(0)_{n\times m}$and let k = 0. Compute the input matrix $B_{n\times n}$= A+I.
- Step 2. Learning

➢ Sub step 2.1.Among all nodes in the network, randomly select a node viwith the input vector $x = B_i$.

➢ Sub step 2.2.For j = 1 to m, calculate $W_j(k)$, the connection between node vi to the output neuron Cj. Calculate the discriminator function η (W j(k),x)by equation (1). Then determine a winner neuron j according to (2).

➢ Sub step 2.3.Update the connection weight matrix W(k+1) by the formulae (3) and (4).

➢ Sub step 2.4.Repeat Substep 2.1 to Substep 2.2 until all nodes are learned.

➢ Sub step 2.5.Update the learning rate parameter aif adaptive learning rates are used. If $\|W(k+1)-W(k)\|<e$ or k >MaxIter, go to Step 3; otherwise, let k = k+1 and go to Sub step 2.1.

- Step 3. Output

Classify all nodes into no more than m groups according to their final winner neurons. Return the corresponding communities of the network. The discriminator function is the normalized correlation [2].

$$\eta(W^j, x) = (W^j \cdot x)^T B(W^j \cdot x) \qquad 1$$

The winner neuron $\bar{C}j$ with respect to the input vector x is selected by the following rule:

$$\bar{j} = \text{argmax}\eta(W\, j, x). \qquad 2$$

The weights associated with the winner neuron are updated as follows

$$W^{\bar{j}}(k+1) = \frac{W(k) + \alpha B_i}{\| W^{\bar{j}}(k) + \alpha B_i \|_\infty} \qquad 3$$

And the weights associated with the non-winner neuron are updated as follows

$$W^{j}(k+1) = \frac{W(k) + \alpha (1 - B_i)}{\| W^{j}(k) + \alpha (1 - B_i) \|_\infty}, \qquad 4$$

Where α is the learning rate, and $j = \bar{j}$. After the training phase, the nodes are mapped into no more than m communities. The communities are constructed according to the final connection weight matrix W. For the i-th node, we define that it belongs to the $\bar{j}$-th community if $\bar{j} = \arg\max w_{ij}$ [2].

III. RESULTS AND ANALYSIS

The basic SOM algorithm was tested by us not on real world networks but on small experimental networks.

3.1 For a Disc Distribution

The input is in the form of a disc in two dimensions along the X and Y axes. The self organising map aligns itself with the input. Attached is the screen shot of 2000 iterations of the disc distribution.

Befor **After**

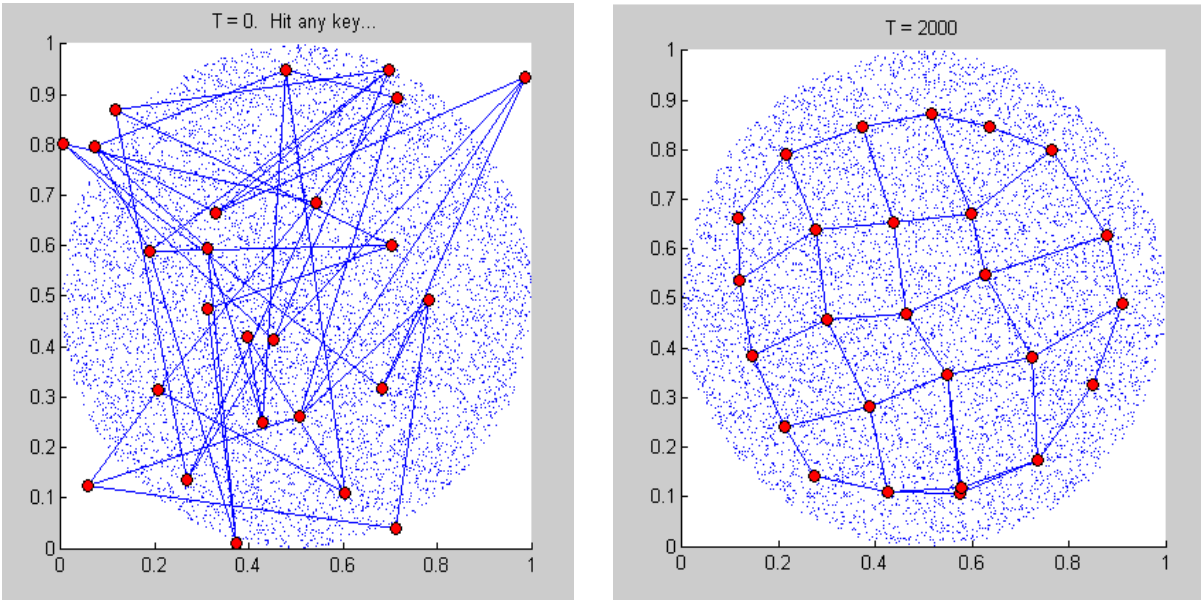

Figure 1. Result of Disc Distribution with 2000 iterations

3.2 For a Square Distribution

Before: **After:**

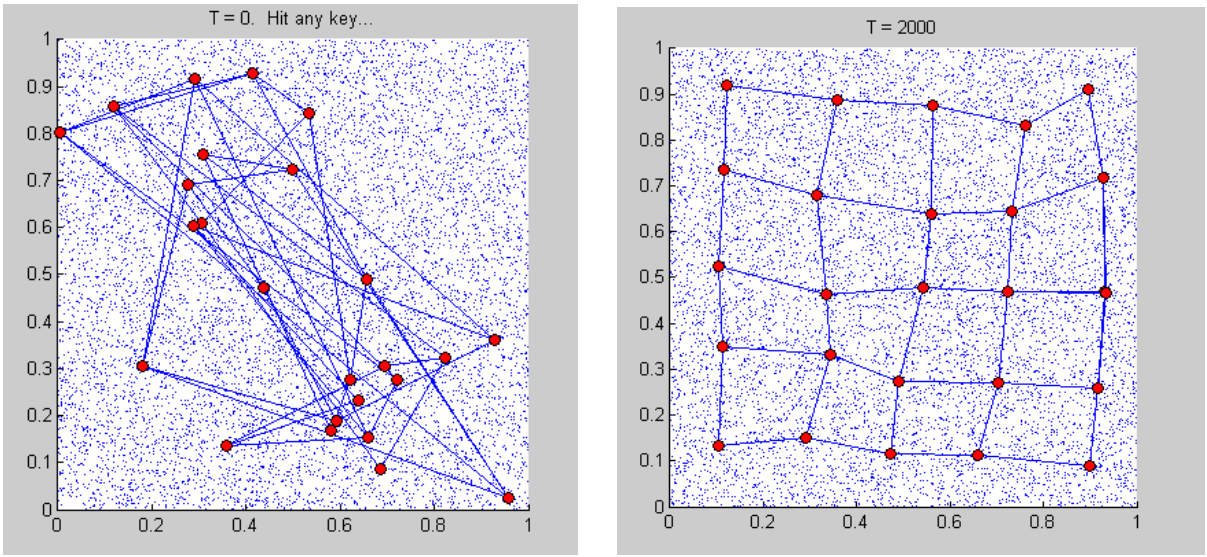

Figure 2. Result of Square Distribution with 2000 iterations

3.3 For a Ring Distribution

Before: After:

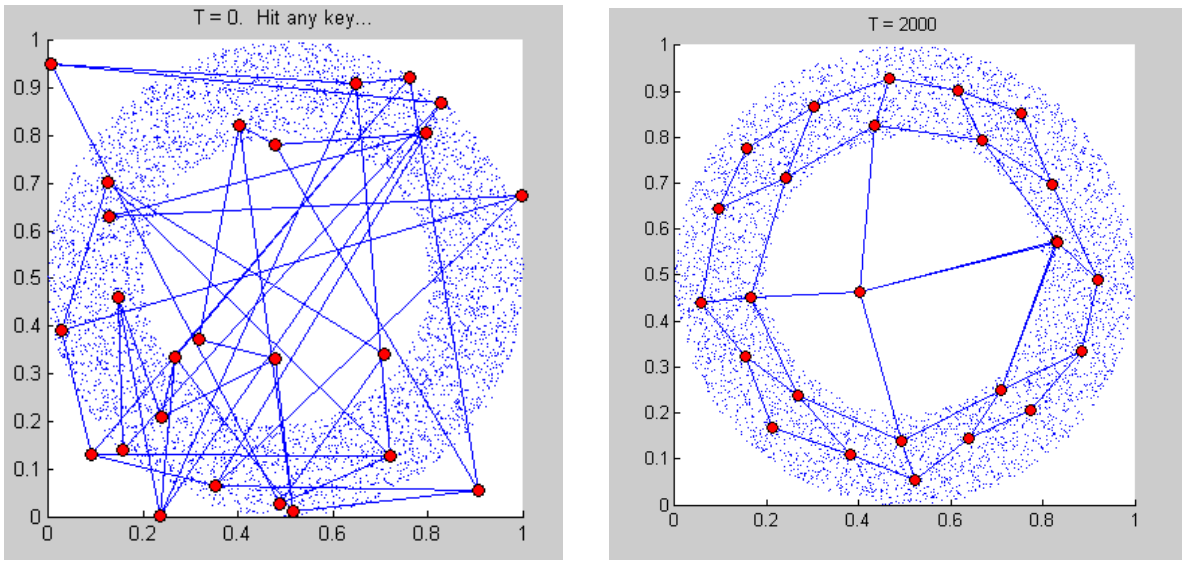

Figure 3. Result of Ring Distribution with 2000 iterations

We first test our method on two small networks depicted in Fig.4, for the small network in Fig. 4(a), our method can detect the two dense sub graphs as communities (denoted by circles). Node 7 can belong to either the left community or the right community. In fact, the connection weight of node 7 to either community is not distinctly larger. Such a case is very common in social networks since some nodes are sparsely connected with other nodes and do not form a community. For the small modular network in Fig.4(b), the three apparent communities can be easily detected by our algorithm, From these two small networks, we can see that our algorithm can efficiently identify the underlying community structure, and especially is able to discard some sparsely connected nodes according to their connection weights.

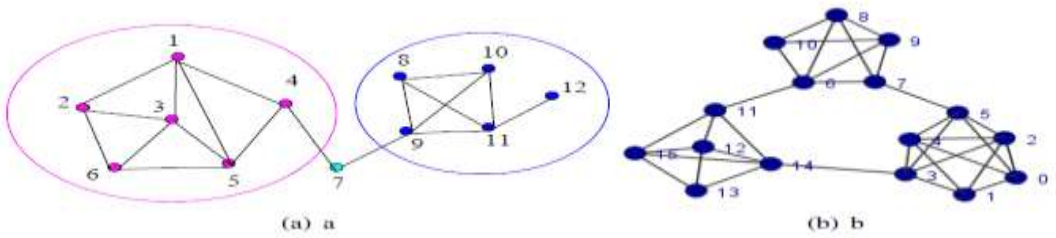

Figure 4. Simple Examples of Modular Social Networks [2]

The performance of our method is a little better than that of spectral algorithm and marginally worse than that of optimization of D. But our method is very fast and its running time for each network is no more than 20 seconds, while the running time of solving the integer programming for optimization D is more than 1 minute. This indicates that our method can be used for large scale networks.

Table 1: Comparison of Modularity values with dataset

DATASETS	GN	MOGA	BOCD	E-SOM
Zachary Karate Club	0.380	0.415	0.419	0.3886
American College Football	0.518	0.515	0.577	0.4857

Table 2: Comparison of Different Modularity Values for Different Communities in Zachary Karate Club Dataset

Number of Communities	Number of Elements in each Community	Modularity
2	29;5	0.0460
3	12;17;5	0.3870
4	12;8;10;4	0.3886
5	4;11;3;11;5	0.1884

Table 3: Comparison of Different Modularity Values for Different Communities in American College Football dataset

Number of Communities	Modularity
10	0.4637
11	0.4857
12	0.4558
13	0.4492

Hence from the above tables we can clearly see that modularity is maximum when there are four partitions of values 12,8,10 and 4 nodes and the maximum value of modularity is 0.3870. The modularity for the American College dataset peaks when the number of partitions is 11 and the maximum value of modularity is 0.4857.

IV. CONCLUSION AND FUTURE WORK

In this paper, we proposed a new community detection algorithm based on self organizing map. It can automatically organize a network into dense sub-graphs without any heuristic manipulation. Besides the

efficiency and effectiveness both on weighted and undirected networks, the self-organizing approach can also identify communities in bipartite networks. In addition, this community detection algorithm is suitable for very large networks without knowing the number of communities. In the future research, we will explore the application of this method to detect communities in biological networks also, we will design more efficient SOM using new operation and weight updating schema.

REFERENCES

[1] J. Golbeck, (2013), Analyzing the Social Web, Morgan Kaufmann, ISBN 978-0-12-405531-5.

[2] Zhenping Li, Rui-Sheng Wang, "Luonan Chen,Extracting Community Structure of complex networks by Self-Organizing Maps", third International Symposium on Optimization and Systems Biology (OSB'09), Zhangjiajie, China, September 20–22, 2009, pp. 48–56.

[3] Duncan Watts, "How The NSA Uses Social Network Analysis To Map Terrorist Networks", 12 June 2013.

[4] Claudio Oscar Dorso, A. D. Medus," Community Detection in Networks", I. J. Bifurcation and Chaos 361-367 (2010)