Statistical Outlook for Community Mining in Social Networks

Aditi Agrawal[1]**, Pawan Prakash Singh**[2]

Gyan Vihar School of Engineering & Technology, Jaipur-302017, India

Abstract: ***Social networking is very popular on the web and the combination with data mining techniques open up more opportunities for social intelligence on. A Social Network can be viewed as a complex interconnection of social entities, comprises of social structure of nodes tied together with one or more type of relationship namely share interests, activities, friendship, background, dislike, trade, financial exchange, etc. Mining a social is the work of grouping these social entities and there patterns for further discrimination, characterization, classification. Much research has been done in the past on social mining algorithm. In this work, we will present a new algorithm Breadth First Droving (BFD) which uses statistical outlook for social mining in Social networks. The algorithm proceeds in breadth first way and incrementally extract communities from the Network. This algorithm can be scaled easily for large Social networks and also fast and simple.***

Keywords: Social network; community mining

1. Introduction

A Social Network comprises of social structure of nodes tied together with one or more type of relationship namely friendship, dislike, trade, financial exchange, etc. Bo Yang et al. [1] defined a social network as a graph G = (V, E), where V = {V_1, V_2, V_3, …. ,V_n} is the set of vertices, and E is the set of edges connecting pairs of vertices. Each edge represents the social relationships among two nodes representing people. They have heterogeneous and multi-relational dataset resubmitted by graphs. Typically these graphs are very large and both nodes and links have attributes. Social networks need not to social in context. There are more real world instances of economic, biological, technological and business social networks.

A social network in its simplest form can be viewed as a weighted bidirectional graph. Positive weights are represented as +1.In the literature, we find a number of weighted graphs in which the weights assigned to the edges lie in a particular range of numbers. However, these graphs may be considered as a special case of the previously explained graph as we can transform these types of graphs to simple graphs by assigning +1.

2. Previous Work

Now days, Social networks become the hot topic because of more researches and usage in this field. A lot of research has been done on social networks in past. In the literature, more algorithms have been developed to detect network communities. They can generally be divided as in [1] into three main categories: 1) *Graph theoretic ways* like Random walk ways and physics-depend ways Spectral ways 2) *Divisive algorithms* like `Betweenness' algorithms of Girvan and Newman[16,17,21], Tyler algorithm and Radicchi algorithm [20] in which they divide the network into smaller subsections. 3) *Agglomerative algorithms* like Modularity-depend algorithms [18] which form communities by joining nodes together.

Girvan and Newman introduced ‘betweenness’ measure in 2002 which iteratively removes edges with the highest “stress” to eventually find disjoint communities. Clauset [18] in 2004 suggested a faster algorithm but the number of droves must still be specified by the user. Flake et al. in 2000 used max-flow min-cut formulation to find communities around a seed node. Kelsic [11] in 2005 introduced an agglomerative algorithm for constructing overlapping communities using local shells, and implement ways for visualizing overlap among communities. Pons and Latapy [12] in 2005 reported a social finding way applying random walk. It starts with single-node communities and again performs the merging of a pair of adjacent communities that minimizes the mean of the squared distances among each node and its social.

Hildrum [10] in 2005 submitted a cut-depend focused social search algorithm. Palla [14] in 2005 used clique percolation for the issue of identifying communities, where one node can belong to more than one social. Their way first identifies all cliques of the network and performs a standard component analysis of the clique-clique overlap matrix to discover a set of k-clique-communities.

Kim and Jeong [15] in 2005 developed a matrix block diagonalization and applied it to weighted stock networks. Their way constructs a network of stocks and identifies stock groups with a percolation outlook depend on a filtered empirical stock correlation matrix.

Newman [9] in 2006 introduced eigen-spectrum of a matrix and calls it modularity matrix, which plays a role in social detection similar to that played by the graph Laplacian in graph partitioning calculations. Qian [6] in 2006 submitted a link mining algorithm to identify communities of practice depend on the idea that linked nodes belonging to the same social should have a larger number of 'common friends'. Ichise[7] in 2006 submitted, a social mining system which assists to find communities of researchers by using bibliography data, in this way the key feature is the modeling of papers and researchers, which enables us to eliminate the edges of large droves.

Yang and Liu [5] in 2006 submitted an incremental force-depend algorithm which allows mining communities in large scale dynamic networks, which is inspired by Newtonian gravitational law; it considers degree of vertex as mass and each edge as virtual spring. Recently Yang et al. [1] in 2007 developed a new algorithm, called FEC, for identifying communities from signed social networks. The key idea behind it rests on an agent-depend random walk model, depend on which the FC phase can find the sink social including a specified node with a linear time complexity. Thereafter, the sink social is extracted from the entire network by the EC phase depend on some robust graph cut criteria. In one other paper by Yang and Liu [2] in 2007 they submitted an Agent -depend AOC outlook to solving Distributed Network Social Mining Issue (D-NCMP), In this outlook, the nodes and links of distributed networks are distributed among a group of autonomous agents, who are responsible for finding all natural communities hidden in distributed networks [3], depend on their respective local views.

3. Problem Statement

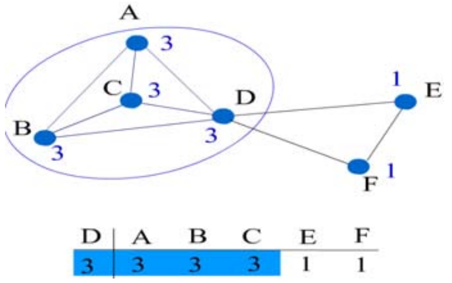
Social network analysis is an emerging research area due to its wide usage. Social mining is the issue to discover the group of nodes that shares similar properties. The issue of identifying communities in a network is usually modeled as graph clustering (GC)[19] issue or subgraph identification issue. More authors worked on this issue and have given various outlooks to mine out social structure. In Chapter 2 we will dissert most of these outlooks in detail. After studying existing algorithms we concluded that these algorithms use complex calculations and concepts to extract communities.

We are concerned to find a solution to mine communities out of social networks efficiently, to lower time complexity giving fast execution with increase in network size and that doesn't need to provide external parameter for droving, i.e., it should be fully automatic.

4. BFD (Breadth First Droving)

4.1 Main Idea

The communities in a network are formed by droving groups of nodes closely connected to each other. The algorithm uses breadth-first traversal, as disserted in Cormen *et al.* [4, 8], as its propagation way. In breadth first traversal all the neighbor nodes in social are traversed first and then nodes of other social. Whenever a node is traversed all its neighbor's visit counter is incremented, when we reach on a node having visit counter 2 or more, it signifies that a drove may exists if majority of its neighbors are traversed more than twice see fig. 1.



D	A	B	C	E	F
3	3	3	3	1	1

Figure 1: Drove Formation

The case may arise for a node if its neighbors belong to more than one class then the vertex is assigned to the class with maximum common class neighbors, which we call as majority of participation rule. For any node V there exist four parameters prior to its classification:

- NAV ← No. of Adjacent Vertices of V.
- UDV ← No. of Undroved Vertices adjacent to V having visitcounter ≥ 2.
- DV ← No. of Droved Vertices adjacent to V having visitcounter ≥ 2.
- MCDN ← Max. Common Drove Number.

Let us have a look on the illustrations shown in fig. 2, the vertex neighbors belong to three droves (7, 6, and 4) and have two undroved neighbors.

We dissert 3 cases that will arise here:

Case 1: NAV/2 > (UDV+DV)
Case 2: UDV > DV considering case 1 is false.
Case 3: DV > UDV considering case 1 is false.

Case 1 arises when number of neighbors with visitcounter < 2 is more than remaining neighbors. In this case the vertex V is left undroved and reconsidered after, as the algorithm proceeds. Case 2 arises when number of undroved neighbors with visitcounter ≥ 2 is greater than remaining neighbors. In this case a new class is formed for vertex V along with its undroved neighbors. Case 3 arises when number of droved neighbors with visitcounter ≥ 2 is greater than remaining neighbors. In this case the class with maximum number of linked neighbors is selected and vertex V is appended into that class. So, the vertex V is assigned to drove 7.

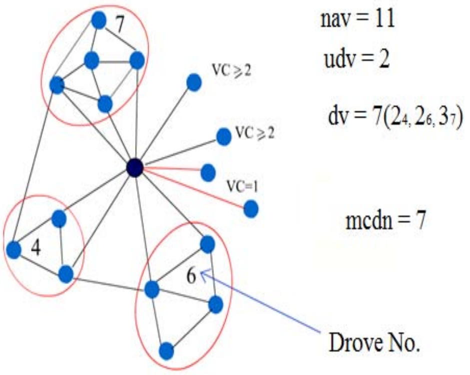


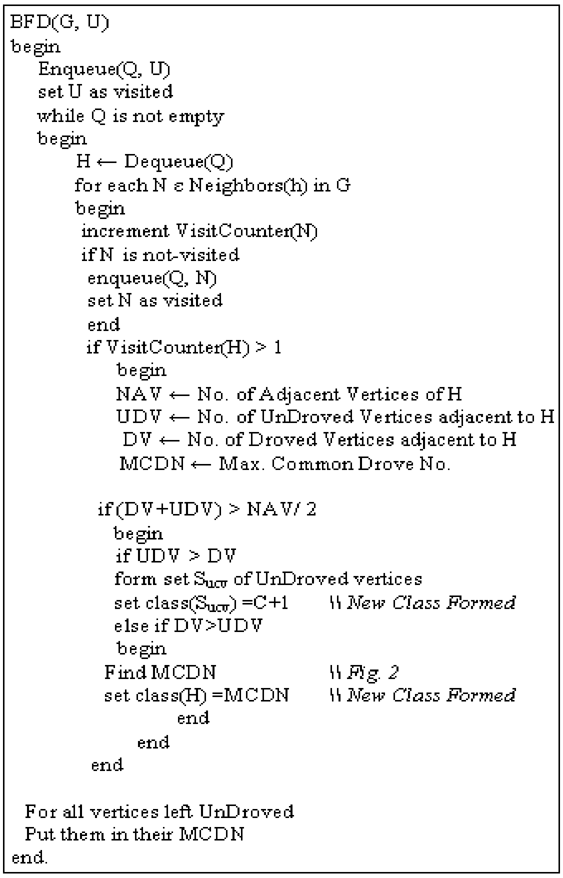
Figure 2: Example showing all parameters

In case there is no single class majority is there the vertex is left undroved, for illustrations in the fig. 2 if the node equally participates in all droves (Drove 7 also has 2 links with it) no single majority lies (so MCDN in that case should be 0).

Small tightly coupled components are detected first which merges nearby vertices together to form big drove on the basis of majority of participation incrementally. The detection of nodes of same social shows flood filling behavior. As the traversal to network grows small components drove together to form communities. The calculation is simple, fast and can be scaled for large networks.

4.2 The Algorithm

Following is pseudo code for the BFD algorithm. The algorithm uses queue data structure is resubmitted by Q having enqueue and dequeue operations.



```
BFD(G, U)
begin
    Enqueue(Q, U)
    set U as visited
    while Q is not empty
    begin
        H ← Dequeue(Q)
        for each N ε Neighbors(h) in G
        begin
         increment VisitCounter(N)
         if N  is not-visited
          enqueue(Q, N)
          set N as visited
          end
         if VisitCounter(H) > 1
             begin
             NAV ← No. of Adjacent Vertices of H
             UDV ← No. of UnDroved Vertices adjacent to H
              DV ← No. of Droved Vertices adjacent to H
             MCDN ← Max. Common Drove No.

           if (DV+UDV) > NAV/ 2
             begin
             if UDV > DV
             form set $S_{ucv}$ of UnDroved vertices
             set class($S_{ucv}$) =C+1       \\ New Class Formed
             else if DV>UDV
             begin
            Find MCDN                   \\ Fig. 2
            set class(H) =MCDN          \\ New Class Formed
                      end
                 end
          end

  For all vertices left UnDroved
  Put them in their MCDN
end.
```

4.3 Algorithm Evaluation

Consider a network illustrations (fig. 3), we start from a randomly chosen vertex V_0, as we traverse the network and update the statics of network (see table 3.1), we will see how communities are detected incrementally out of the network.

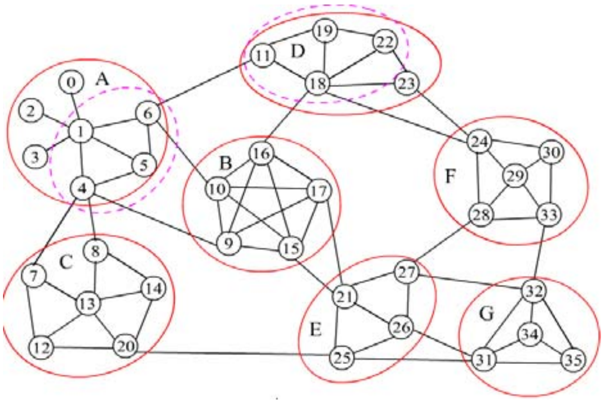


Figure 3: A Social Network Illustration

	V(VC)	Adjacent Vertices of V(VisitCounter)	
1	0(0)	1(1)	
2	1(1)	0(1),2(1),3(1),4(1),5(1),6(1)	
3	2(1)	1(2)	
4	3(1)	1(3)	
5	4(1)	1(4),5(2),7(1),8(1),9(1)	
6	5(2)	1(5),4(2),6(2)	A
7	6(2)	1(6),5(3),10(1),11(1)	
8	7(1)	4(4),12(1),13(1)	
9	8(1)	4(5),13(2),14(1)	
10	9(1)	4(6),10(2),15(1),16(1),17(1)	
11	10(2)	6(2),9(2),15(2),16(2),17(2)	B
12	11(1)	6(3),18(1),19(1)	
13	12(1)	7(2),13(3),20(1)	
14	13(2)	7(3),8(2),12(2),14(2),20(2)	C
15	14(2)	8(3),13(4),20(3)	
16	15(2)	9(3),10(3),16(3),17(3),**21(1)**	
17	16(3)	9(4),10(4),15(3),17(4),18(2)	
18	17(4)	9(5),10(5),15(4),16(4),21(2)	
19	18(2)	11(2),16(5),19(2),**22(1),23(1),24(1)**	
20	19(2)	11(3),18(3),22(2)	D
21	20(3)	12(3),13(5),14(3),25(1)	
22	21(2)	15(5),17(5),25(2),**26(1),27(1)**	
23	22(2)	18(4),19(3),23(2)	
24	23(2)	18(5),22(3),24(2)	
25	24(2)	18(6),23(3)**,28(1),29(1),30(1)**	
26	25(2)	20(4),21(3),26(2),**31(1)**	E
27	26(2)	21(4),25(3),27(2),31(2)	
28	27(2)	21(5),26(3),28(2),**32(1)**	
29	28(2)	24(3),27(3),29(2),**33(1)**	F
30	29(2)	24(4),28(3),30(2),33(2)	
31	30(2)	24(5),29(3),33(3)	
32	31(2)	25(4),26(4),32(2),**34(1),35(1)**	
33	32(2)	27(4),31(3),33(4),34(2),35(2)	G
34	33(4)	28(4),29(4),30(3),32(3)	
35	34(2)	31(4),32(4),35(3)	
36	35(3)	31(5),32(5),34(3)	

🟨 New drove formed in v(vc) column.
🟦 Merged with existing drove in v(vc) column.

5. Conclusion and Scope of Future Work

In this thesis work we have disserted Social Networks from research point of view and submitted a new algorithm BFD (Breadth first droving) which uses statistical outlook for social mining in Social networks. The algorithm works in as breadth first way covering breadth of community and increasingly finds them from the Network. This algorithm can be scaled for large Social networks and it is very simple as well as fast. The effectiveness of this outlook has been validated using implementation in GUESS tool.

The time complexity of the algorithm is O(V+E) where V represent number of nodes in the vertices and E represent edges of social network. The algorithm doesn't need any parameter to be supplied for its operation like drove size or number (k) as in more other algorithms and it doesn't encompass complex iterative calculation of measures as in cut depend outlooks.

So far we have tested this algorithm with medium sized networks, in future the algorithm can be enhanced to deal with large and dynamic networks of order higher than 10^5. We haven't touched the multi-relationship view of Social

Networks, so this idea can be extended to cover it. This idea can also be extended to deal with overlapping communities in social networks by using fuzzy system.

References

[1] Bo Yang, W.K. Cheung, and Jiming Liu, Social Mining from Signed Social Networks, IEEE / KDE, VOL. 19, NO. 10, 2007.

[2] Bo Yang An Autonomy Oriented Computing (AOC) Outlook to Distributed Network Social Mining, IEEE/ SASO 2007.

[3] Ying Zhou, Joseph Davis, Discovering Web Network in the Blog space, 40th Hawaii International Conference, 2007.

[4] S. Boccaletti, V. Latora, Y. Moreno, M. Chavez, and D.U. Hwang, Complex networks: Structure and dynamics, *Physics Reports* 424, 175–308 (2006).

[5] Bo Yang, D.Y. Liu, Force Depend Incremental Algorithm for mining social structure in Dynamic Network, J. Comp. Sci. & tech., Vol 21, No.3, May 2006,

[6] Rong Qian, Wei Zhang ,Bingru Yang, Detect social structure from the Enron Email Corpus Depend on Link Mining, ISDA, 2006.

[7] Ryutaro Ichise, Hideaki Takeda, A mining way of communities keeping tacit knowledge, IEEE/ICDMW'06.

[8] Mohsen and Hassan, Different Aspects of Social Network Analysis, IEEE/ WI'06.

[9] M. E. J. Newman, Finding social structure in networks using the eigenvectors of matrices 0605087v3, 2006.

[10] Deng Cai, Zheng Shao, Mining Hidden Social in Heterogeneous Social Networks, *LinkKDD'05*, 2005 ACM.

[11] Eric D. Kelsic, Understanding complex networks with social-finding algorithms, SURF 2005.

[12] P. Pons and M. Latapy, Computing Communities in Large Networks Using Random Walks, Proc. 20th Int'l Symp. Computer and Information Sciences (ISCIS '05), pp. 284-293, 2005.

[13] G. Palla, I. Derenyi, I. Farkas, and T. Vicsek, Uncovering the Overlapping Social Structure of Complex Networks in Nature and Society, Nature, no. 7042, pp. 812-816, 9 June 2005.

[14] D.-H. Kim and H. Jeong, Systematic Analysis of Group Identification in Markets, Physical Rev. E, vol. 72, 046133, 2005.

[15] Jordi Duch, Alex Arenas, Social detection in complex networks using Extremal Optimization, cond-mat/0501368 v1, 2005.

[16] M. E. J. Newman and M. Girvan, Fast algorithm for detecting social structure in networks. Physical Review E-69:026113, 2004.

A. Clauset, M. E. J Newman & C. Moore, Finding social structure in very large networks, Physical Review E 70(066111), 2004.

[17] Andreas Noack, An energy model for visual graph clustering, *GD 2003*, pages 425.436. Springer-Verlag, 2004.

[18] Gary William Flake, Robert E. Tarjan, and Kostas Tsioutsiouliklis, Graph Clustering and Minimum Cut Trees, Internet Mathematics Vol. 1, No. 4: 385-408, 2004.

[19] F. Radicchi, Defining and Identifying Communities in Networks, PNAS, vol. 101, no. 9, pp. 2658-2663, 2004.

[20] M. E. J. Newman, The structure and function of complex networks, *SIAM Review* 45, 167–256 (2003).

About Author

Aditi Agrawal was born in October 10, 1992 in District Firozabad of Uttar Pradesh (India). She received her B. Tech. degree in Information Technology from Suresh Gyan Vihar University, Jaipur (India) in the year 2012. After graduation, in 2013 she obtained Post Graduate in Information Communication from the same university.

Pawan Prakash Singh received his B. Tech. degree from GE U, Dehradun, H.N.B.G.U., Uttarakhand, India and M. Tech. degree from SGVU, Jaipur, India. Presently, he is a Professor with the Computer Engineering Department in Gyan Vihar School of Engineering. & Technology, Jaipur, India. His research interests include Multimedia Broadcasting, Video-on-Demand, Networking, Image processing, Modeling & Segmentation, and Wavelet Applications. He has published many papers in the national and international conferences and journals.