Community Detection by Leader driven algorithm in real world Complex networks

Prajwal M P
Information Technology
National Institute of Technology, Karnataka
Surathkal, India
prajwalmp.192it015@nitk.edu.in

Bavya Balakrishnan
Information Technology
National Institute of Technology, Karnataka
Surathkal, India
bavyabalakrishnan.192it003@nitk.edu.in

***Abstract*—A new trending direction in the research of detecting community in large scale complex networks is Leader driven community detection algorithms. First we identify certain specific vertices as leader nodes in the network. Then we identify local communities around these leader nodes. Through the results we show that the algorithm proposed here identifies communities in networks that are simliar to the real structure of communities present in the network. We have also compared our results with Newmann, Lovain and walktrap algorithms and show that our algorithm performs better. We also explore the topological properties some random graphs.**

***Index Terms*—Community detection, Leader driven algorithms**

I. Introduction

There is a set of topological characteristics that differentiate Large-scale complex networks from pure random graphs. They are, low degree of separation indicating that nodes are more reachable to each other, a higher clustering coefficient indicating that they are grouped and also follow power law degree distribution. In almost all the real world complex networks the nodes tend to be organized to groups called Communities as a result of this. A connected subgraph having nodes similar with one another than with it is with the rest of the the graph is referred to as a community. As the vertices that are considered to be within a community can have some similarities in behaviour we can try to understand more about the networks by discovering and doing an analysis of their main community structure. For example, in an e-commerce site with network of transactions, a community can be considered as customers with similar preferences. A community is the group of pages on the same topic in Web as a complex network.

To perform various tasks such as distribution of computation, graph visualization on a large scale and compression of huge graphs in real-world complex networks we can make use of the community structure that is within them. [1].

Many algorithms have been put forward for community detection in real world networks.

Disjoint community detection - Partitioning the graph or network in such a way one that one vertex can be identified to be a part of at most only one community. Disjoint community detection is the task most of the papers in community detection deals with.

Overlapping community detection - Soft clustering of the graph where a vertex can be a part of many communities at a time.

Local community detection - Instead of clustering the whole graph into communities we compute the community of a given node.

First and second type of algorithms are NP hard problems. Modularity which was initially proposed in [2] is the most popular graph partition criteria in research today. Although few critical drawbacks of approaches based on modularity optimization have been discovered by many researchers. This led to more research for other new emerging techniques for community detection such as label propagation [3] and seed centric approaches [4].

II. Literature review

We have gone through different emerging community detection algorithms and community evaluation approaches as part of survey.

A. Algorithms for Community detection

The four Main classes of Algorithms for Community detection are: Approaches based on Group, network, propagation and seed centric approaches.

1) Group based approaches: Here we identify the set of highly connected nodes that share some strong connectivity similar to the below ones,

- **High connectivity -** This is an NP hard problem. We can consider maximal clique or a gamma quasi clique as a community. A gamma quasi clique is a sub-graph where degree of Graph $\leq \gamma$.

- **High reachability -** Here a maximal n clique sub-graph can be considered as a community structure. A n clique is a sub-graph with the diamteter $\leq$ n. These methods have polynomial Computational complexity.

2) Network based -: The pattern of connection in the large-scale network is considered in this approach. One of the drawback of this type of algorithm is that it takes as input the number of communities to be discovered. The approaches based on optimizing the quality metric are the most popular network approaches. Among all of them modularity is the common one.

If $M = \{D_1, ..., D_k\}$ is a clustering of the vertices set N of a given graph then the modularity of M is computed as

$$R(M) = \sum_{D \epsilon M} e(D) - a(D)^2 \tag{1}$$

where

$$e(D) = \frac{\sum_{k \epsilon D} \sum_{k \epsilon D} A_{kl}}{2 \times m_G} \tag{2}$$

is part of edges that are connecting vertices within the community D, and

$$a(D) = \frac{\sum_{k \epsilon D} \sum_{l \epsilon N} A_{kl}}{2.m_G} \tag{3}$$

is the ratio of edges that are on a vertex of D.

For improving modularity optimisation different heuristic approaches have been discovered.

- **Agglomerative methods -** Algorithm begins by taking each node as a community in this bottom up approach. Then, it progresses by combining some communities based on some quality metric through many iterations. For example, Louvain algorithm [6]. The algorithm has two phases. It initially, by local modularity optimization it identify small communities. and then, it combines vertices of the communities to form a another network where each vertex is considered as a community and then start merging those vertices(communities) to form another set of communities. If the modularity can be increased then those two communities are combined. Until a threshold is reached the steps are repeated.The computing complexity of the approach is O(Vlog(V)).
- **Separative methods -** This is a top-down method. This approach begins by taking the complete network as one subgraph. For partitioning of the network into groups it iterates to find the ties based on various metric. The most known algorithm of this type is Newman-Girvan algorithm [7]. The algorithm is based on the relatively high betweenness centrality shown by a tie connecting different communities as it would be joining together a large number of shortest paths between vertices in various communities. We choose the tie with the largest betweenness centrality at the end of every iteration. This algorithm then loops for m times and gives an output of highest modularity. Since it has large complexity: $O(m^2n + (m)^3log(m))$ this is not apt for large scale real world networks.

Other methods like genetic and evolutionary algorithms can also be used for modularity optimization based methods.

Implicit assumptions made by all modularity optimization approaches are as follows:

- The highest modularity result is the best one.
- The parition with highest possible modularity can be idenitified always if the graph contains a hidden community structure
- There can be structurally similar Partitions with high modularity values if a graph has an underlying community structure.

However, these assumptions do not hold necessarily in complex networks according to recent studies.

Critical drawbacks like these, of modularity optimization algorithms, have resulted in the investigation for more better methods.

3) Propagation based methods -: A label ln is given to each vertex n $\in$ V in graph. Synchronously all nodes update their labels to the most common label in their neighborhood in many iterations. When no node change its label the convergence is attained. Overall computation complexity is O(op) with o iterations for convergence and complexity of each iteration O(p).

Drawbacks

- Convergence to a stable state is not guaranteed
- Less robust, as tie breaking randomly creates different clustering output for different iterations.

4) Seed centered methods: Local communities are identified based on seed nodes in the input network [8], [9], [10]. Algorithm is composed of three main step

[a] Selecting Seed nodes.

[b] Assigning local community for each seed.

[c] Identifying a new group based on the set of local subgraphs estimated before.

Algorithm A : Seed based detection

Input: A connected graph G with set of vertices V and edges E,

1: Initialize (C) to be an empty set
2: Seed = Seed_Selection(G)
3: for k ϵ Seed do
4: K k $\leftarrow$ local_community_assign(k,G)
5: K $\leftarrow$ K + K k //Finding set of local communities
6: end for
7: return community_detection(C)

Algorithm B Leader Driven Community Detection

Input: A connected graph G with set of vertices V and edges E

1: Initialize Leader Set as an empty set #set of leaders
2: for v ϵ V do
3: if isALeader (v) then
4: Add v to LeaderSet
5: end if
6: end for

7: Communities= groupComumunitiesLeader(LeaderSet)
8: for v ϵ V do
9: for c ϵ Communities do
10: membership degree of each node for each community
11: Mem_Degree[v, c] = membership_computation(v, c)
12: end for
13: sorted ranklist for each vertex
14: R[v] = RankSort(Mem_Degree[v])
15: end for
16: repeat the following process
17: for v ϵ V do Adjust the community membership ranklist by merging it with that of direct neighbours
18: RR [v] = AggregateRanklist xϵv
19: R[v] = RR [v]
20: end for
21: until convergence of RR[v] for all v
22: for v ϵ V do
23: /* community assignment task */
24: for comm ϵ R[v] do
25: if Mem_Degree[v,comm] - Mem_Degree[v,R[0]]$\leq \epsilon$.
26: COMMUNITY(c) = COMMUNITY(c) + v
27: end if
28: end for
29: end for
30: return Communities

Leader driven algorithms is a seed centred method. There are 2 categories of nodes: leaders and followers. Leaders are representatives of communities. Followers are assigned to most fitting communities in the assignment step. There are many vertex classification techniques and vertex assignment approaches.

In [11], a method based from the K means clustering algorithm is proposed by authors. Number of partitions is the input to algorithm. A major drawback of the approach is clearly this. Algorithm starts by randomly selecting k nodes as learders. Others are labelled as followers. An community is represented by its leader. We assign each follower node to the leader which is most similar. The follower is identified as outlier if no community is found. Algorithm forms a set of new leader nodes again by selecting the central node as a leader for each community. The process is repeated until stabilization is achieved. The properites of first selected k leaders determines the pace of convergence. Picking the K nodes having highest degree centrality and those have less neighbors in common is considered as the best approach for selection of initial set.

An algorithm that is somewhat similar to our approach is proposed in [10] . A node with less closeness centrality than at least one neighbor can be chosen as a leader. The reciprocal of the sum of distance from the node to all nodes in the graph gives closeness centrality. In descending order of closeness centrality the leaders are placed. Every leader is provided followers that are not yet assigned to any other nodes in leader role. Every leader with no followers can be assigned to the most common community chosen by most of its neighbours.

B. Community based methods -

Three main classes of approaches for evaluation:

1. Approach for the networks with ground truth communities information.

2. Approach considering the topological properties of communities estimated.

3. Task-based evaluation.

1) Ground truth community comparison methods -: To obtain ground truth community structure for networks we have many ways.

Experts: Experts in the similar area of work can define the ground truth community for some small graphs. Different properties of the network such as the distribution of degree, size, density, clustering-coefficient, the size of communities and community separation degree can be decided by user in [12]

generators : Constructing artificial networks for a given community structure is the idea here.

Using Implicit community structure : With the help some implicit characteristics and semantics of partition structure. Authors of [13] identifies set of fans of a particular artist as a community in Live journal social-network .

When a ground truth of partition is provided, we can use normal clustering rating metrics to evaluate the methods. Here we use mainly two metrics namely, the Adjusted Rand-Index [14] and Normalized Mutual Information [15].

ARI metric is the count of elements that are grouped in same groups in two communities being compared. Let the two partitions vertex set be given as Q i = Q i 1 ,. . . . . , Q i l , Q j =Q j 1 , ... , Q j k . Now the groups are given by

- Set 11 is the vertices are within same groups in both Q i and Q j
- Set 00 is the vertices are within different groups in both Q i and Q j
- Set 10 is the vertices which are in same groups within Q i are in different groups within Q j
- Set 01 is the vertices which are in different groups within Qi are in same groups in Q j

The rand-index is given by:

$$R(Q_i, Q_j) = \frac{2 \times (n_{11} + n_{00})}{n \times (n-1)} \tag{4}$$

ARI is the Normalized difference of the Rand-Index.

$$ARI(Q_i, Q_j) = \frac{\sum_{x=1}^{i} \sum_{y=1}^{k} \left(\begin{array}{c} \mid Q_i^x \cap Q_j^y \mid \\ 2 \end{array} \right) - t_3}{\frac{1}{2}(t_1 + t_2) - t_3} \tag{5}$$

where,

$$t_1 = \sum_{x=1}^{l} \left(\begin{array}{c} \mid Q_i^x \mid \\ 2 \end{array} \right), t_2 = \sum_{y=1}^{k} \left(\begin{array}{c} \mid Q_i^y \mid \\ 2 \end{array} \right), t_3 = \frac{2t_1 t_2}{n(n-1)} \tag{6}$$

For dissimilar groups this metric gives zero whereas for similar ones this gives 1. From what is known about about

partition Qj how efficiently we cluster a randomly picked element from V in a partition Qj is put into mutual information.

$$H(Q_i) = -\sum_{x=1}^{l} \frac{\mid Q_i^x \mid}{n} log_2 \left(\frac{\mid Q_i^x \mid}{n} \right) \quad (7)$$

,

the Shannon's entropy of a partition Qi

$$MI(A,B) = H(A) + H(B) - H(A,B) \quad (8)$$

gives the mutual information of two variables
a normalized version given in [**?**]:

$$NMI(A,B) = \frac{MI(A,B)}{\sqrt{H(A)H(B)}} \quad (9)$$

III. Methodology

A. The Leader Driven community Detection

The algorithm of the approach is given in Algorithm B

1.Function is Leader () : The Leader node is supposed to possess greater centrality vallues. In our experiments we have explored 3 centrality measures.

Degree centrality:Degree centrality is the ratio of vertices directly connected to the given vertex.

Betweenness centrality: For a given vertex its betweenness centrality is computed by counting number of shortest paths going through the vertex for every other vertex pairs out of all the possible shortest paths that are connecting them.

Eigenvector-centrality :It is centrality measure which takes into account how important a vertex connected to the current vertex is along with number of connections.

If the measure of centrality is higher than or equal to $\sigma \in$ [0, 1] percent of its neighbors centralities it is identified as a leader. High value of σ means less number of leaders.

2.Function computecommunitiesleaders If the fraction of common members compared to all of neighbors is above a given value say $\delta \in$ [0,1], then the two vertices are assigned to same community. Number of communities depends on σ and δ.

3.Function memebership(v, c) Degree of Membership a node n and community c is given by,

$$mem(n,c) = \frac{1}{(min_{x \epsilon COMM(c)} ShortestPath(n,x)) + 1} \quad (10)$$

Membership of all leaders of a community is 1 in that community.

4. Rank aggregation methods

If we are given a group of community membership preference list given by vertices,each vertex will aggregate its list by combining this with preference lists of its immediate neighbors in the raph. Different strategies can be used for this aggregation.

position-based approaches: Borda's Method [20]. If $M = [M_1, M_2, M_3, ..., M_k]$ is the ranked list then the Bordas score of element j in M_k is,

TABLE I: Topological characteristics of networks

Datasets	No of Nodes	No of Edges	No of Real communities
Zachary's	34	78	2
Football	115	616	12
US-Politics	100	411	3
Dolphins	62	159	2

$$BM_k(j) = count(i) \mid M_k(i)M_k(j) \& i \in M_k. \quad (11)$$

$B(j) = \Sigma_{t=1} B_{L_t}(j)$,is the total score taking into account all lists order based methods - Kemeny optimal [21]. Almost all proposed kemeny methods use a normal sorting algorithm after doing comparison like Ai is given more importance compared to Aj if most of rankers prefer Ai over Aj.

5.Community assignment Assignment is based on the highest membership value of each vertex. Inorder to bring overlapping a node is given to partitions for which its value is ϵ-different from the value to the highest ranked.

IV. Experimentation and Results

A. Evaluation on benchmark networks

We used igraph graph analysis toolkit in R to implement the algorithm. In order to evaluate the performance of this proposed approach we selected four broadly used benchmark networks. We made sure that ground truth community decomposition is available for this networks. A brief description about each network is given below.

Zachary karate club This is a network found in US university's Karate club with 34 members back in 1970 [16]. After an argument between administrator and instructor of the club the club got divided into 2 groups and the instructor ended up creating a club by himself with his own followers.

American college football dataset This is a network where each node represent American football team [17]. teams playing games get an edge between them. According to the count of the games played among them the teams are partitioned to 12 groups with each containing 8 to 12 members. This count of games will also depend on the distance.

American political-books [22] It is a copurchasing network of online seller Amazon.com about the books related to US politiccs sold by them. Each vertex constitutes a book. Based on the copurchasing by a given buyer edges are created among books. Based on the reviews that were posted on Amazon about the books Mark Newman segregated the books into 3 three disjoint groups such as liberal, conservtive and neutral.

Dolphins network [18]. An undirected network created based on the behaviour of a community containing 62 dolphins observed over a period of 7 years. If frequent assosications between pair of dolphins is observed then edges are added to join them. It is divided into two partitions.

The characteristic of selected networks and their Ground truth communities given by the color is shown in (1). (I) gives Basic properties of networks.

By making changes in the configuration parameters as follows we applied the LdCd algorithm on each network.

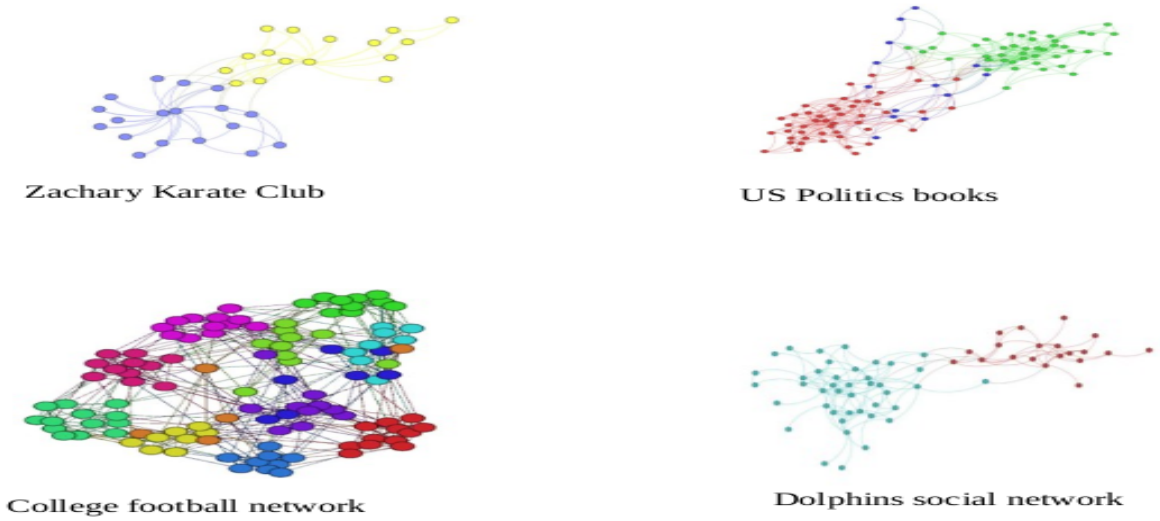

Fig. 1: Ground truth of networks

- Centrality metrics are Betweenness-centrality (BC), Degree-centrality (dc) and Eigen-Vector Centrality
- Voting method for Rank aggregation is Local Kemeny and Borda
- $\sigma \in$ [.5, .6, .7, .8, .9, 1.0]
- $\delta \in$ [.5, .6, .7, .8, .9, 1.0]
- $\epsilon \in$ [.0, .1, .2]

We compute the evaluation metrics such as NMI, ARI and the modularity Q for each of the above given configurations. The results obtained from different configurations are shown in (II).

Variation in σ has significant impact on the results obtained. Fixing σ at 0.8 or 0.9 gave best results. This observation supports the idea of introducing the σ parameter and not always considering the vertex as leader if it has top most centrality in its direct neighborhood. ϵ has negligible impact on obtained results. We observed decreased values of NMI and ARI with increase in ϵ threshold. As the ground truth community structure of all the selected networks are disjoint introducing overlapping degree with higher values of ϵ adversely affect the performance of algorithm. Reducing δ decreased the number of communities generated for certain datasets and give good values for NMI,ARI. Since it is not logically accepted for all complex networks we fix 0.9 and 0.5 as the best values for δ. We observe that the variation in NMI, ARI and Q has a close dependency with the centrality and the voting method in the configuration choice.

In the experimets we conducted we observe that the betweenness centrality provides more convergence for towards the correct community structure to get. Though Eigen value centrality is expected to give good results it doesn't work best for the network we chose. Borda out performs Local Kemeny in all the networks.

Community structure of the four selected benchmark networks obtained by proposed LdCd by the best 2 parameter configurations are shown in (2) and (3)

We identified that the performance of a configuration varies with networks and this has close relation to the specialities and topology of network. The dependency between configuration parameters and speciality of network is topic to be explored deeper.

We also tested the same set of networks with other advanced algorithm and compared the results, The Newman Girvan algorithm [7], WalkTrap [19] and the Louvain [6]. The configuration choice for LdCd is the following: Centrality metric is betweenness centrality, Voting method is Borda, σ = 0.9 δ = 0.9/0.5 , and ϵ= 0. (III) gives results on the 4 datasets by using different algorithms.

The results obtained clearly reveal that proposed algorithm outperforms other algorithms for Zachary,Dolphins and US Politics networks. We obtained good performance in the football network as well. This could be because of the by the absence of leaders in the network, making the communities detection task more difficult.

We can conclude that modularity metric does not correspond to the best decomposition into communities as measured by both NMI and ARI. To be precise, we always observe the best modularity for Louvain method(even better than the modularity of the ground-truth decomposition), however, it is not ranked first according to NMI. Parameter configurations with high values of σ gives the best results in our approach. We also applied the algorithm on a large scale Email network generated by a large European research institution. An edge is formed between person u and person v if u sent v at least one mail. There are 1005 nodes and 25571 edges in the network. Each person is a member of any one of 42 departments at the institute. Due to topological property of graph we observed we could only obtain average performance for it in comparison to

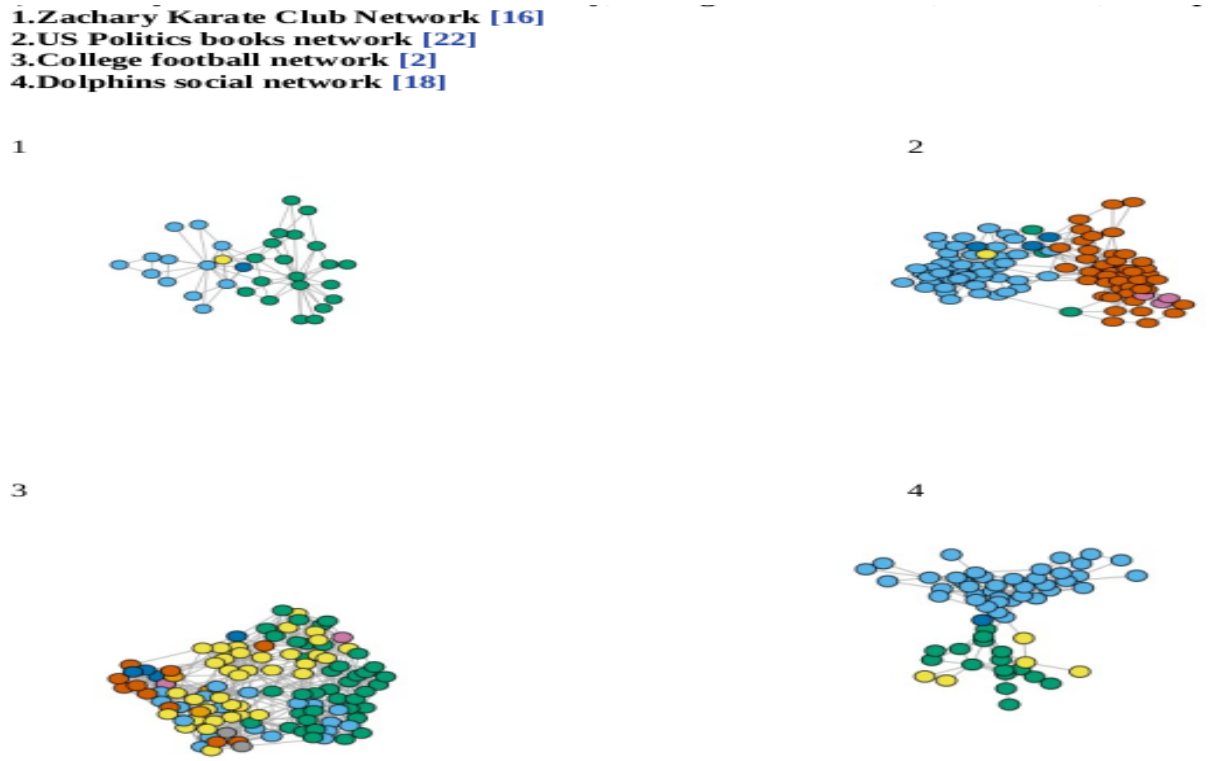

Fig. 2: Community structure detected by LdCd for the 4 benchmark networks with configuration (Centrality metric:betweenness centrality, Voting method:Borda, $\sigma = \delta = 0.9$,and Epsilon = 0)

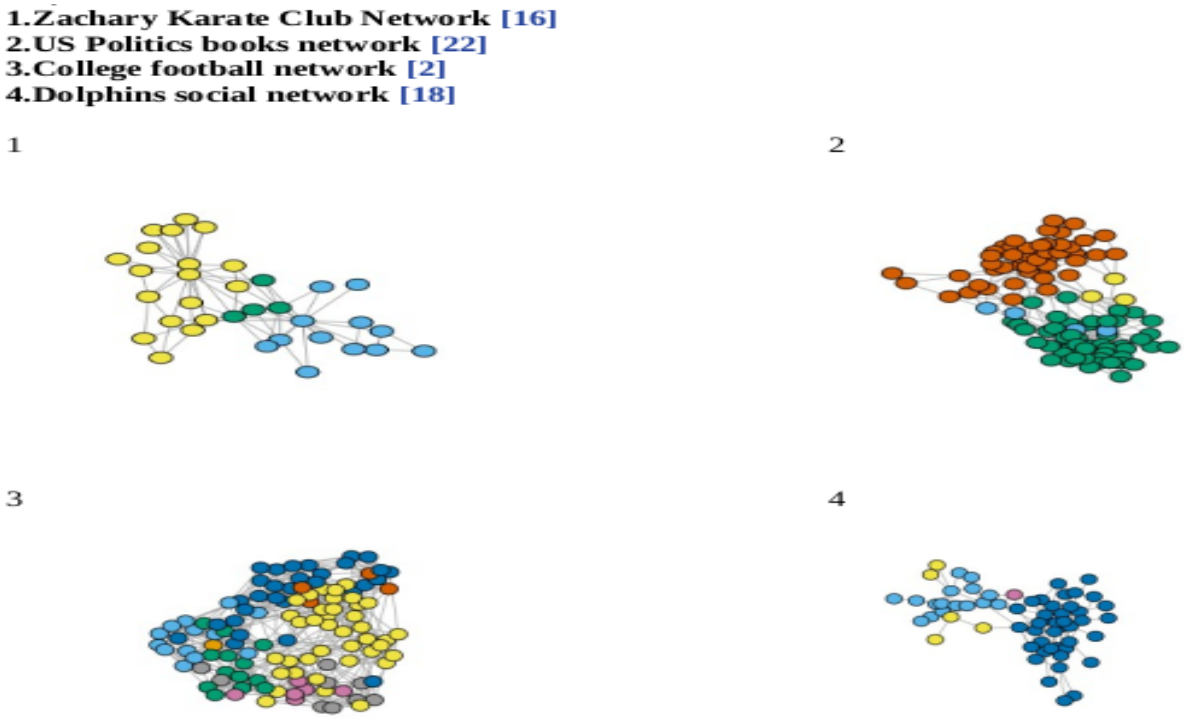

Fig. 3: Community structure detected by LdCd for the four benchmark networks with configuration (Centrality metric:betweenness centrality, Voting method:Borda, $\sigma = 0.9$, $\delta = 0.5$,and Epsilon= 0)

TABLE II: Comparison of performances of applying LdCd to different configuration parameters

Datasets	Algorithm	NMI	ARI	Modularity	No Communities
Zachary	Sigma=0.9,delta=0.9				
	LICOD(Borda_Betweenness)	0.6337324	0.6819667	0.3051446	4
	LICOD(Borda_EigenVector)	0.6337324	0.6819667	0.3051446	4
	LICOD(Borda_Degree)	0.6964905	0.745851	0.4222855	4
	LICOD(Kemeny_Betweenness)	0.6140008	0.523254	0.2635602	5
	LICOD(Kemeny_EigenVector)	0.3699431	0.2801601	0.1938692	6
	LICOD(Kemeny_Degree)	0.3699431	0.2801601	0.2048817	6
	Sigma=0.9,delta=0.5				
	LICOD(Borda_Betweenness)	0.839372	0.8279669	0.3391683	3
	LICOD(Borda_EigenVector)	0.7209632	0.7530574	0.3258547	3
	LICOD(Borda_Degree)	0.839372	0.8279669	0.3391683	3
US Politics	Sigma=0.9,delta=0.9				
	LICOD(Borda_Betweenness)	0.5181528	0.6076945	0.4228922	6
	LICOD(Borda_EigenVector)	0.5181528	0.6076945	0.4228922	6
	LICOD(Borda_Degree)	0.5135687	0.5956606	0.4222855	6
	LICOD(Kemeny_Betweenness)	0.3583441	0.2938903	0.244029	11
	LICOD(Kemeny_EigenVector)	0.3907821	0.2802048	0.1969498	17
	LICOD(Kemeny_Degree)	0.3844464	0.3493627	0.1854166	13
	Sigma=0.9,delta=0.5				
	LICOD(Borda_Betweenness)	0.5448607	0.6486401	0.4447247	4
	LICOD(Borda_EigenVector)	0.5448607	0.6486401	0.4447247	4
	LICOD(Borda_Degree)	0.5448607	0.6486401	0.4447247	4
Football	Sigma=0.9,delta=0.9				
	LICOD(Borda_Betweenness)	0.5393786	0.2122235	0.285424	14
	LICOD(Borda_EigenVector)	0.5185883	0.1987756	0.2740979	15
	LICOD(Borda_Degree)	0.5303016	0.240639	0.3555429	11
	LICOD(Kemeny_Betweenness)	0.5394264	0.1460565	0.08332247	38
	LICOD(Kemeny_EigenVector)	0.5165736	0.1244962	0.06052655	41
	LICOD(Kemeny_Degree)	0.5187954	0.156331	0.0945847	30
	Sigma=0.9,delta=0.5				
	LICOD(Borda_Betweenness)	0.5281761	0.221265	0.3019661	12
	LICOD(Borda_EigenVector)	0.4948808	0.2001475	0.3063052	13
	LICOD(Borda_Degree)	0.5479841	0.2649123	0.392897	9
Dolphins	Sigma=0.9,delta=0.9				
	LICOD(Borda_Betweenness)	0.7688508	0.8430689	0.3470393	4
	LICOD(Borda_EigenVector)	0.7688508	0.8430689	0.3470393	4
	LICOD(Borda_Degree)	0.8089225	0.8846558	0.3442506	3
	LICOD(Kemeny_Betweenness)	0.4107441	0.4223021	0.2359875	6
	LICOD(Kemeny_EigenVector)	0.6533707	0.5416906	0.2994541	5
	LICOD(Kemeny_Degree)	0.412597	0.4047125	0.2772833	6
	Sigma=0.9,delta=0.5				
	LICOD(Borda_Betweenness)	0.7688508	0.8430689	0.3470393	4
	LICOD(Borda_EigenVector)	0.7688508	0.8430689	0.3470393	4
	LICOD(Borda_Degree)	0.7174538	0.8258898	0.3517068	3

other advanced algorithms. The results are tabulated in (IV)

B. Task Driven Evaluation using Data clustering

We explore the task driven evaluation of Community detection algorithms by converting data clustering task to community detection. There are some works already been published with the above idea [23]. The approach is as described below

Step 1. Choose some sparse classical datasets for experiment. We have selected 5 public datasets from UCI website [24]. Their properties are given in (VI) .

Step 2. Compute an n*n similarity/distance matrix for the n datapoints. We can use various measures as given in Table V for estimating distance.

Step 3. Construct Relative neighborhood graph (RNG) by connecting data items xi,xj if they satisfies the below rule.

$$d(x_i, x_j) \leq max_l\{d(x_i, x_l), d(x_j, x_l)\}, \forall l \neq i, j \qquad (12)$$

Step 4. Apply the proposed algorithm for community detection on the graph constructed. Apply other top state of the art algorithms as well.

Step 5. Using various evalution criteria compare the performances of algorithms

We have tabulated the features of obtained graphs in (VI).

Using different distance measures we constructed RNG graphs on all the selected datasets and results are shown in (VIII).

We observe some real world network characteristics such as small diameter and low density on these graphs. Unfortunately the graphs obtained by using Chebyshev distance shows high density. However, graphs has high clustering coefficient. Graphs shows high transitivity as well.

On the Relative neighbourhood graphs obtained by applying Cosine similarity we tested the community detection algorithms as they are relatively closer to real world networks. Results obtained on all 4 RNG graphs applying 4 different

TABLE III: Performance Comparison of 4 community detection algorithms

Dataset	Algorithm	NMI	ARI	Modularity	#Communities
Zachary	Newman	0.5798278	0.4686165	0.4012985	5
	Louvain	0.5866348	0.4619069	0.4188034	4
	Walktrap	0.504178	0.3331266	0.3532216	5
	LICOD(Borda_Betweenness)	0.839372	0.8279669	0.3391683	3
US Politics	Newman	0.5584515	0.6823684	0.5168011	5
	Louvain	0.512133	0.5579848	0.5204853	4
	Walktrap	0.5427476	0.6534224	0.5069724	4
	LICOD(Borda_Betweenness)	0.5448607	0.6486401	0.4447247	4
Football	Newman	0.8788884	0.7781023	0.599629	10
	Louvain	0.8903166	0.8069409	0.6045696	10
	Walktrap	0.8873604	0.8154427	0.6029143	10
	LICOD(Borda_Betweenness)	0.5281761	0.221265	0.3019661	12
Dolphins	Newman	0.5541605	0.3949115	0.5193821	5
	Louvain	0.5108534	0.3274327	0.5185317	5
	Walktrap	0.53725	0.416739	0.4888454	4
	LICOD(Borda_Betweenness)	0.7688508	0.8430689	0.3470393	4

TABLE IV: Performance of LdCd on large scale network

Dataset	Algorithm	NMI	ARI	Modularity	#Communities
email-Eu-core network	Louvain	0.5361345	0.2505032	0.4211831	27
	Walktrap	0.580412	0.1974976	0.3501706	145
	LICOD(Borda_Betweenness)	0.2035174	0.01793908	0.06871058	46

Distance	Formula
Euclidean distance	$dist_{euc}(x,y) = \sqrt{\sum_{i=1}^{n} \mid xi - yi \mid^2 2}$
Cosine similarity	$dist_{cos}(x,y) = 1 - \frac{x.y}{\mid x \mid \mid y \mid}$
Chebyshev distance	$dist_{cheb}(x,y) = max_i(x_i - y_i)$

TABLE V: Distance measures used.

TABLE VI: Structural properties of Datasets used for RNG

Dataset	**Glass**	**Iris**	**Wine**	**Vehicle**	**Abalone**
No of datapoints	214	150	178	846	4177
No of attributes	10	5	14	19	8
No of classes	7	3	3	4	29

community detection algorithms are then compared using different evaluation criteria such as ARI, NMI and modularity and they are tabulated in (VII). The actual class information for each data item is already available in the dataset. We observe that modularity is not a good criteria for quality measurement. The highlighted values indicate the better performance of LdCd over other algorithms. For wine and Abalone the LICOD outperforms other algorithms. It shows good results for other datasets as well.

V. Conclusion and Future work

We explored a new trending approach of leader driven community detection algorithm in large scale complex networks. Its capacity to identify communities is demonstrated by 4 widely used benchmark social networks. We also proposed a new task-driven community evaluation approach using data clustering. As a future development to the proposed approach we think about improving the speed of algorithm to perform better in multiplex and bipartite networks.

References

[1] Grabowski, S., Bieniecki, W.: Tight and simple web graph compression. CoRR, abs/1006.0 (2010)

[2] Girvan, M., Newman, M.E.J.: Community structure in social and biological networks. PNAS 99(12), 7821–7826 (2002)

[3] Raghavan, U.N., Albert, R., Kumara, S.: Near linear time algorith to detect community structures in large-scale networks. Phys. Rev.E 76, 1–12 (2007)

[4] Kanawati, R.. Seed-centric approaches for community detection in complex networks. In: Meiselwitz, G., (ed.) 6th International Conference on Social Computing and Social Media, volume LNCS 8531, pp. 197–208, Crete, Greece. Springer, New York (2014)

[5] Verma, A., Butenko, S.: Graph partitioning and graph clustering. In: Bader, D.A., Meyerhenke, H., Sanders, P., Wagner, D. (eds.) Network clustering via clique relaxations: a community based approach. Contemporary Mathematics, pp. 129–140. American Mathematical Society, Providence (2012)

[6] Blondel, V.D., Guillaume, J.-L., Lefebvre, E.: Fast unfolding of communities in large networks. J. Stat. Mech. Theory Exp. 2008, P10008 (2008)

[7] Newman, M.J., Girvan, M.: Finding and evaluating community structure in networks. Phys. Rev. E 69, 02613:1–022613:15 (2004)

[8] Kanawati, R.: LICOD: Leaders identification for community detection in complex networks. In: SocialCom/PASSAT, pp. 577–582 (2011)

[9] Papadopoulos, S., Kompatsiaris, Y., Vakali, A.: A graph-based clustering scheme for identifying related tags in folksonomies. In: DaWak, pp. 65–76 (2010)

[10] Shah, D., Zaman, T.: Community detection in networks: The leader-follower algorithm. In: Workshop on Networks Across Disciplines in Theory and Applications, NIPS (2010)

[11] Khorasgani, R.R., Chen, J., Zaiane, O.R.: Top leaders community detection approach in information networks. In: 4th SNA-KDD Workshop on Social Network Mining and Analysis, Washington D.C. (2010)

[12] Lancichinetti, A., Radicchi, F.: Benchmark graphs for testing community detection algorithms. Phys. Rev. E 4, 046110 (2008)

[13] Yang, J., Leskovec, J.: Defining and evaluating network communities based on ground-truth. In: Zaki, M.J., Siebes, A., Yu, J.X., Goethals, B., Webb, G.I., Wu, X. (eds.) ICDM, pp 745–754. IEEE Computer Society (2012)

[14] Hubert, L., Arabie, P.: Comparing partitions. J. Classif. 2(1), 192–218 (1985)

[15] Strehl, A., Ghosh, J.: Cluster ensembles: a knowledge reuse framework for combining multiple partitions. J. Mach. Learn. Res. 3, 583–617 (2003)

[16] Zachary, W.W.: An information flow model for conflict and fission in small groups. J. Anthropol. Res. 33, 452–473 (1977)

TABLE VII: Performance comparison of LdCd with Walktrap, Louvain, Newman–Girvan algorithms

Dataset	Algorithm	NMI	ARI	Modularity	No of communities
Iris	Newman	0.6894212	0.4709471	0.7228967	8
	Louvain	0.5890717	0.3748021	0.7187097	9
	Walktrap	0.6562614	0.4987374	0.6972523	12
	LICOD	0.6565191	0.5399218	0.4503184	2
Glass	Newman	0.4575755	0.2138958	0.7670461	11
	Louvain	0.476197	0.217018	0.7563275	12
	Walktrap	0.4179286	0.1384216	0.7403119	16
	LICOD	0.4597846	0.3052357	0.3798834	5
Wine	Newman	0.308765	0.1534895	0.8134986	13
	Louvain	0.3159939	0.1362786	0.7944163	12
	Walktrap	0.3017275	0.09450665	0.7867851	17
	LICOD	0.3683113	0.3543734	0.4127416	2
Vehicle	Newman	0.2383895	0.11754	0.796356	14
	Louvain	0.2331711	0.1143421	0.7868221	14
	Walktrap	0.2467031	0.1158469	0.7698765	16
	LICOD	0.1360652	0.07975364	0.4266861	2

TABLE VIII: Topological properties of Relative neighbourhood graphs

Dataset	Property	Euclidean	Chebyshev	Cosine
Iris	**No of Edges**	404	2476	442
	Diameter	33	14	23
	Average degree	5.38666666666667	33.0133333333333	5.89333333333333
	Density	0.036152125279642	0.221565995525727	0.039552572706935
	Transitivity	0.08252688172043	0.348853912013634	0.033080808080808
Glass	**# Edges**	558	7786	552
	Diameter	21	8	24
	Average degree	5.21495327102804	72.7663551401869	5.1588785046729
	Density	0.024483348690273	0.341626080470361	0.024220086876399
	Transitivity	0.013966480446927	0.252269101340897	0.011970534069982
Wine	**# Edges**	380	514	438
	Diameter	102	84	59
	Average degree	4.26966292134831	5.7752808988764	4.92134831460674
	Density	0.024122389386149	0.03262870564337	0.02780422776614
	Transitivity	0	0.178455284552845	0
Vehicle	**# Edges**	2602	4072	2774
	Diameter	63	54	45
	Average degree	6.15130023640662	9.62647754137116	6.55791962174941
	Density	0.007279645250185	0.011392281114049	0.007760851623372
	Transitivity	0.004284490145673	0.091105407372458	0

[17] Girvan, M., Newman, M.E.J.: Community structure in social and biological networks. PNAS 99(12), 7821–7826 (2002)

[18] Lusseau, D., Schneider, K., Boisseau, O.J., Haase, P., Slooten, E., Dawson, S.M.: The bottlenose dolphin community of doubtful sound features a large proportion of long-lasting associations. Behav. Ecol. Sociobiol. 54, 396–405 (2003)

[19] Pons, P., Latapy, M.: Computing communities in large networks using random walks. J. Graph Algorithms Appl. 10(2), 191–218 (2006)

[20] Borda, J.C.: Mémoire sur les élections au scrutin. Comptes rendus de l'Académie des sciences, traduit par Alfred de Grazia comme Mathematical Derivation of a election system, Isis, vol. 44, pp. 42–51 (1781)

[21] Kemeny, J.G.: Mathematics without numbers. Daedalus 88, 571–591 (1959)

[22] Krebs, V. Political books network. http://www.orgnet.com

[23] de Oliveira, T.B.S., Zhao, L., Faceli, K., de Carvalho, A.C.P.L.F.:Data clustering based on complex network community detection.In: IEEE Congress on Evolutionary Computation, 2008 (CEC2008), 1-6 June 2008, Hong Kong, pp. 2121–2126 (2008)

[24] http://archive.ics.uci.edu/ml/datasets.html.