Department of IT, NITK Surathkal [Dec-Jun 2020]

WSC (IT752) - Mini-project Progress Evaluation - I

Leader-driven community detection

Bavya Balakrishnan -192IT003
Prajwal M P -192IT015

16-Apr-2020

Paper chosen for implementation

LICOD: A Leader-driven algorithm for community detection in complex networks

Zied Yakoubi · Rushed Kanawati

Agenda

- Introduction
- Literature Survey
- Outcome of Literature Survey
- Problem Statement
- Objectives
- Methodology
  - Proposed enhancements/novelty
- Work done and Implementation
- Experimental Results and analysis
  - Results of Innovative work done
- Individual Contribution
- Conclusion and Future Work

Introduction

- **Leader-driven community detection algorithms for community detection in large-scale complex networks.**
- Complex Networks share a set of non-trivial characteristics that distinguish them from pure random graphs such as
  - Low separation degree (or what is better known as small-world feature)
  - Power law distribution of node’s degrees
  - High clustering coefficient
- Real world complex networks exhibit a level of organization, called communities.
- A community is defined as a connected subgraph whose nodes are much linked with one each other than with nodes outside the subgraph.

  eg. In web as a complex network, a community would be a set of pages dealing with a same topic

- Useful in computation distribution,huge graph visualization and large-scale graph compression
- Different types of community detection algorithms are there:
  - *Disjoint communities detection*: compute a partition of the graph node’s set where one node can belong to only one community at once
  - *Overlapping communities detection* : A node can belongs to several communities at once
  - *Local community identification* : to compute the community of a given node rather than partitioning the whole graph into communities

Introduction

- Disjoint and overlapping community detection problems are NP-hard
- Most applied graph partitioning criteria are the modularity.
- Serious limitations of modularity optimization-based approaches boosted the alternative approaches
- Emergent approaches include label propagation approaches and seed-centric ones
- The basic idea of seed-centric approaches is to select a set of nodes (i.e. seeds) around which communities are constructed.
- Special case of seeds is to select nodes that are likely to act as leaders of their communities
- since LdCD algorithms are not based on maximizing an objective function (i.e. the modularity),we use **Task-oriented evaluation.**
- We propose a method to transform classical clustering benchmarks into a community detection problem

Literature Review

Authors	Methodology	Advantages	Limitations
Tsironis, S., Sozio, M., Vazirgiannis, M	Using Network based approach considering whole connection patterns in the network.	Spectral clustering approaches and hierarchical clustering approaches canbe used.	● Number of clusters to be found should be provided as an input for the algorithm. ● High computation complexity which might be cubic on the size of the input dataset ● All modularity optimization approaches make implicit assumptions which do not hold,
Raghavan, U.N., Albert, R., Kumara, S.	Label propagation approach.All nodes update in a synchronous way their assigned labels by selecting the most frequent label in the direct neighborhood.	A low complexity incremental approaches for community detection	number of iterations grows in a logarithmic way with the growth of target network size
Ovelgönne, M.: ASONAM. In: Rokne, J.G., Faloutsos, C.	distributed label propagation	Fast algorithm handling very large-scale networks	● First, there is no formal guarantee of the convergence to a stable state. ● Lastly, it lacks for robustness, since different runs produce different partitions due to random tie breaking.

Literature Review

Authors	Methodology	Advantages	Limitations
Cordasco, G., Gargano, L	Asynchronous, and semi-synchronous label updating	hinder the problem of oscillation and improve convergence conditions	● these approaches harden the parallelization of the algorithm by creating dependencies among nodes ● they increase the randomness in the algorithm making the robustness even worse.
Khorasgani, R.R., Chen, J., Zaiane, O.R	inspired from the K-means clustering algorithm Leader driven seed centric approach	Most relevant communities are obtained	● algorithm requires as input the number k of communities to identify. ● The convergence speed depends on the quality of initially selected k leaders.

Outcome of Literature Review

- All modularity optimization approaches make implicitly the following assumptions:
  - The best partition of a graph is the one that maximize the modularity.
  - It is possible to find a precise partition with maximal modularity if a network has a community structure
  - Partitions inducing high modularity values are structurally similar if a network has a community structure.
- All three above-mentioned assumptions do not hold necessarily in complex networks according to recent studies.
- These serious drawbacks of modularity-guided algorithms have boosted the research for alternative approaches.
- Leader-driven algorithms constitute a special case of seed-centric approaches.
- Different algorithm follow different node classification approaches and different node assignment strategies.
- Here, we propose a ldCd algorithm insptred by recent seed centric approaches.

Problem Statement

- A new trend in community detection in large-scale complex networks is Leader-driven community detection algorithms. Basic idea here is to identify some particular nodes in the target network, called leader nodes, around which local communities can be computed . We propose an algorithm for leader driven community detection and evaluate its performance on benchmark networks and also using task driven evaluation.

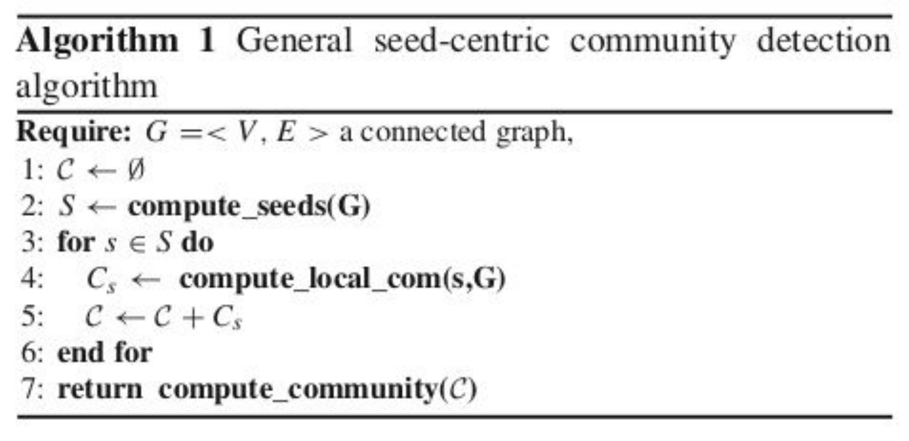
Research Objectives

- Implementing the Leader-driven community detection algorithm and evaluating its performance with different configuration parameters on various networks.
- Evaluating the performance of algorithm using different centrality measures.
- Testing the iterative version of the algorithm on large-scale networks
- Introducing task-oriented evaluation of community detection algorithms and providing an approach for evaluating different community detection algorithms on data clustering tasks.

Methodology

Seed-centric approaches

- The basic idea underlying seed-centric approaches is to identify some particular nodes in the target network, called seed nodes, around which local communities can be computed



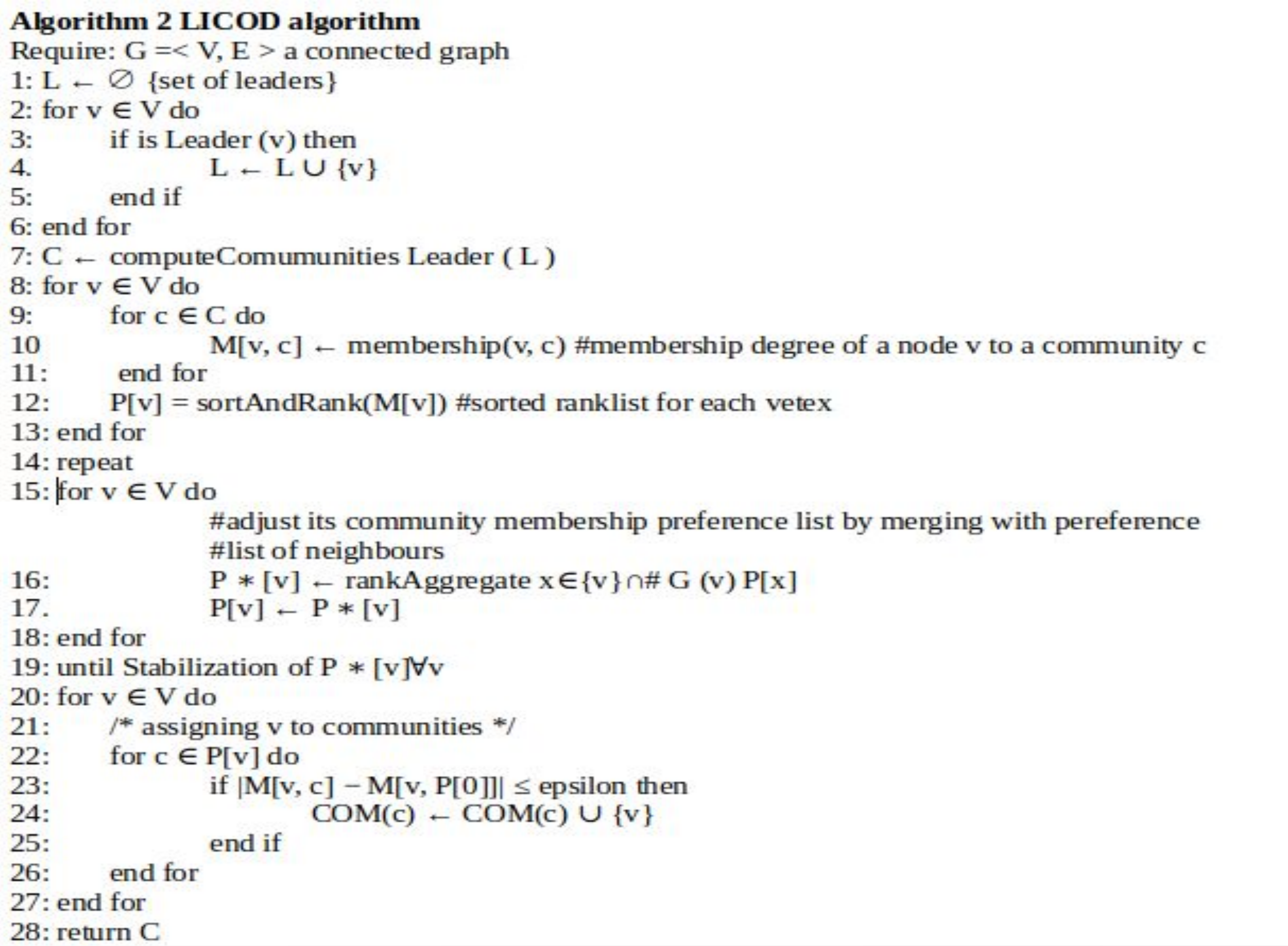
Algorithm 1 General seed-centric community detection algorithm

```
Require: $G =< V, E >$ a connected graph,
$\mathcal{C} \leftarrow \emptyset$
$S \leftarrow$ compute_seeds(G)
for $s \in S$ do
    $C_s \leftarrow$ compute_local_com(s,G)
    $\mathcal{C} \leftarrow \mathcal{C} + C_s$
end for
return compute_community($\mathcal{C}$)
```

- Leader-driven algorithms constitute a special case of seedcentric approaches
- Nodes of a network are classified into two (eventually overlapping) categories: leaders and followers.

Methodology

- Leaders represent communities.



Algorithm 2 LICOD algorithm

```
Require: $G =< V, E >$ a connected graph
$L \leftarrow \emptyset$ {set of leaders}
for $v \in V$ do
        if is Leader (v) then
                $L \leftarrow L \cup \{v\}$
        end if
end for
$C \leftarrow$ computeComumunities Leader ( L )
for $v \in V$ do
        for $c \in C$ do
                $M[v, c] \leftarrow$ membership(v, c) #membership degree of a node v to a community c
         end for
        P[v] = sortAndRank(M[v]) #sorted ranklist for each vetex
end for
repeat
for $v \in V$ do
                #adjust its community membership preference list by merging with pereference
                #list of neighbours
                $P * [v] \leftarrow$ rankAggregate $x \in \{v\} \cap \# \; G \; (v) \; P[x]$
                $P[v] \leftarrow P * [v]$
end for
until Stabilization of $P * [v] \forall v$
for $v \in V$ do
        /* assigning v to communities */
        for $c \in P[v]$ do
                if $|M[v, c] - M[v, P[0]]| \leq$ epsilon then
                        $COM(c) \leftarrow COM(c) \cup \{v\}$
                end if
        end for
end for
return C
```

Methodology

- **Algorithm is implemented using the igraph graph analysis toolkit**

Function is Leader () :

Based on nodes centralities

- Degree centrality
- Betweenness centrality
- Eigenvector centrality

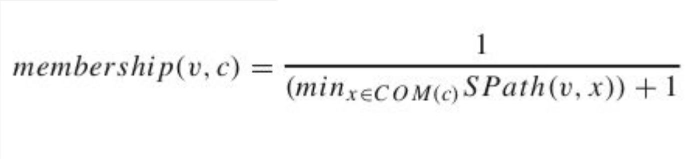
A node is identified as a leader if its centrality is greater or equal to σ ∈ [0, 1] percent of its neighbors centralities.

Function computeCommunitiesLeaders

Two leaders are grouped in the same community if the ratio of common neighbors to the total number of neighbors is above a given threshold δ ∈ [0, 1].

Methodology

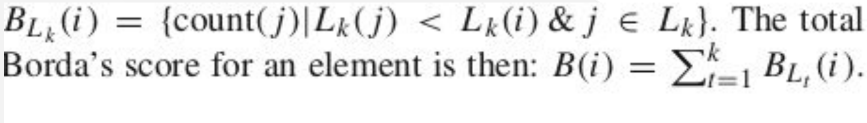
Function membership(v, c)



$$membership(v, c) = \frac{1}{(min_{x \in COM(c)} SPath(v, x)) + 1}$$

Rank aggregation approaches

- Requirement:minimum number of pairwise disagreements
- Borda’s method



$B_{L_k}(i) = \{\mathrm{count}(j) | L_k(j) < L_k(i) \,\&\, j \in L_k\}$. The total Borda’s score for an element is then: $B(i) = \sum_{t=1}^{k} B_{L_t}(i)$.

- Kemeny optimal aggregation

si is preferred to sj ,if the majority of rankers ranks si before sj

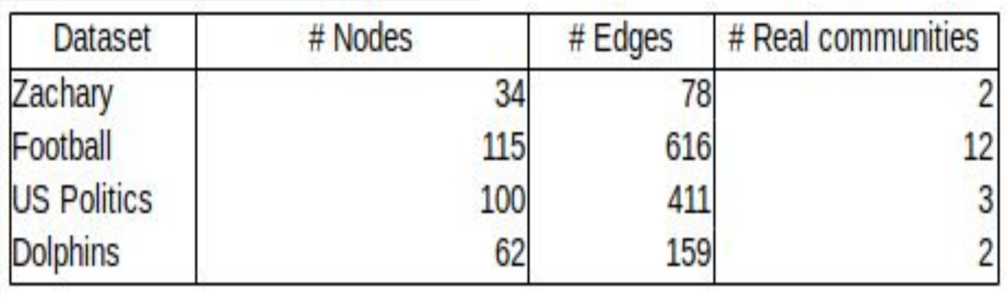
Community assignment

- Each node will be assigned to top-ranked communities in its final obtained membership preference list.
- Threshold epsilon controls the degree of desired overlapping

Methodology

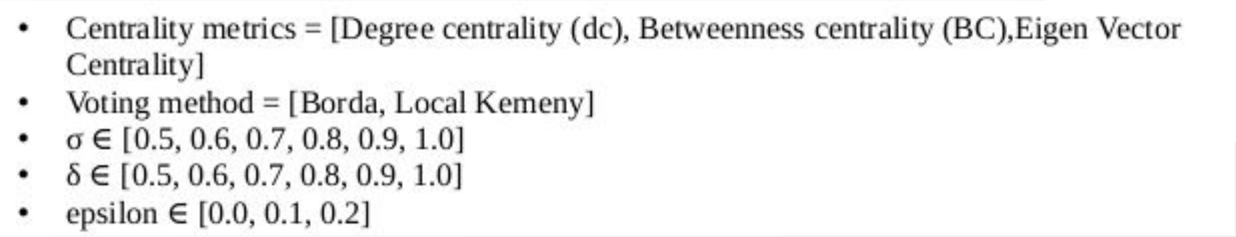
Datasets

The proposed approach is evaluated on a set of four widely used benchmark networks for which a ground-truth decomposition into communities is known.



Dataset	# Nodes	# Edges	# Real communities
Zachary	34	78	2
Football	115	616	12
US Politics	100	411	3
Dolphins	62	159	2

Configuration parameters



- Centrality metrics = [Degree centrality (dc), Betweenness centrality (BC),Eigen Vector Centrality]
- Voting method = [Borda, Local Kemeny]
- $\sigma \in [0.5, 0.6, 0.7, 0.8, 0.9, 1.0]$
- $\delta \in [0.5, 0.6, 0.7, 0.8, 0.9, 1.0]$
- epsilon $\in [0.0, 0.1, 0.2]$

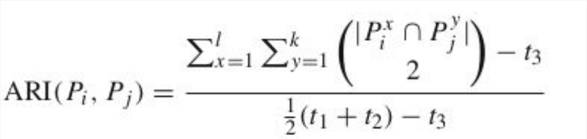
Methodology

Community evaluation criteria

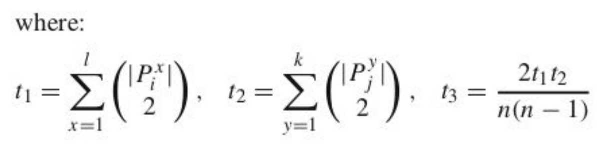
When a ground-truth community structure is available, we can use classical external clustering evaluation indices to evaluate and compare community detection algorithms.

- **Adjusted Rand Index (ARI) :**

  $P\,i = \{P\,i_1 , . . . , P\,i_l\}$, $P\,j = \{P\,j_1 , . . . , P\,j_k\}$ be two partitions of a set of nodes V .



$$\mathrm{ARI}(P_i, P_j) = \frac{\sum_{x=1}^{l} \sum_{y=1}^{k} \binom{|P_i^x \cap P_j^y|}{2} - t_3}{\frac{1}{2}(t_1 + t_2) - t_3}$$



where:

$$t_1 = \sum_{x=1}^{l} \binom{|P_i^x|}{2}, \quad t_2 = \sum_{y=1}^{k} \binom{|P_j^y|}{2}, \quad t_3 = \frac{2 t_1 t_2}{n(n-1)}$$

- **Normalized Mutual Information (NMI)**

  We seek to quantify how much we reduce the uncertainty of the clustering of randomly picked element from V in a partition Pj if we know Pi .

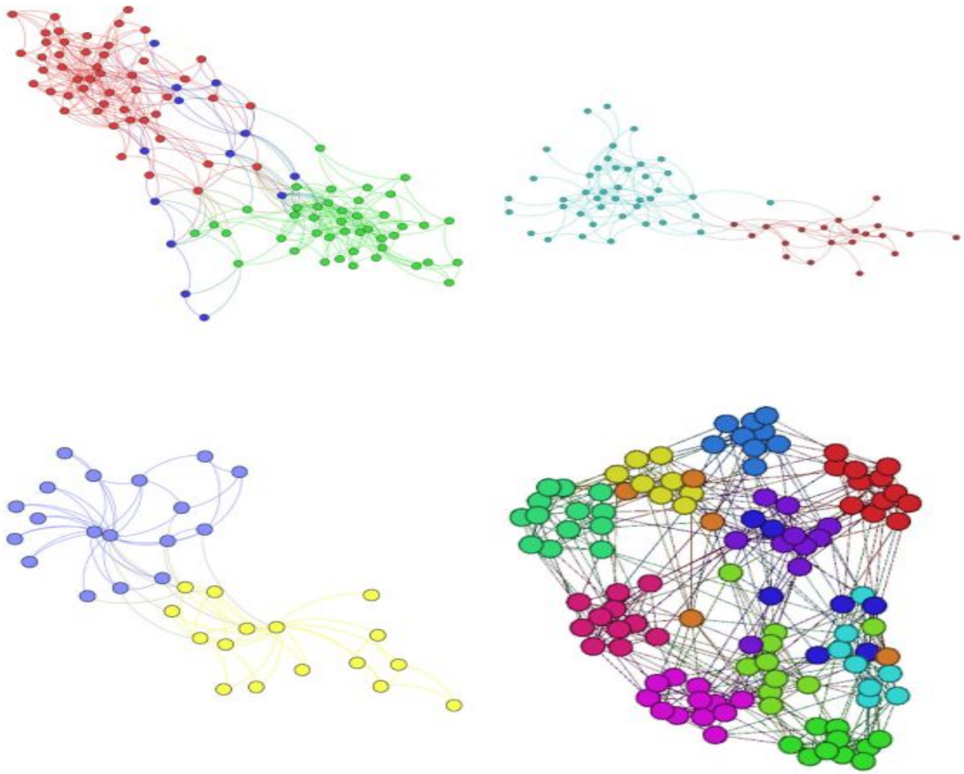
Proposed enhancements/novelty

- For each network we apply the proposed algorithm by changing the configuration parameters.
- We apply new centrality measure Eigenvector centrality for leader identification and analyse its impact in the proposed algorithm.
- Try with different approximate Kemeny aggregation approaches for rank aggregation
- Test the algorithm with different rank aggregation approaches
- Test the algorithm algorithm on large-scale networks
- Introduce an Iterative version of algorithm to ensure the stabilization of communities obtained

Work done and Implementation

- We evaluated the proposed approach on a set of four widely used benchmark networks for which a ground-truth decomposition into communities is known.
- We went through the basic topological characteristics and real community structure of four selected networks.

1.Zachary Karate Club
2.College football network
3.US Politics books network
4.Dolphins social network



Work done and Implementation

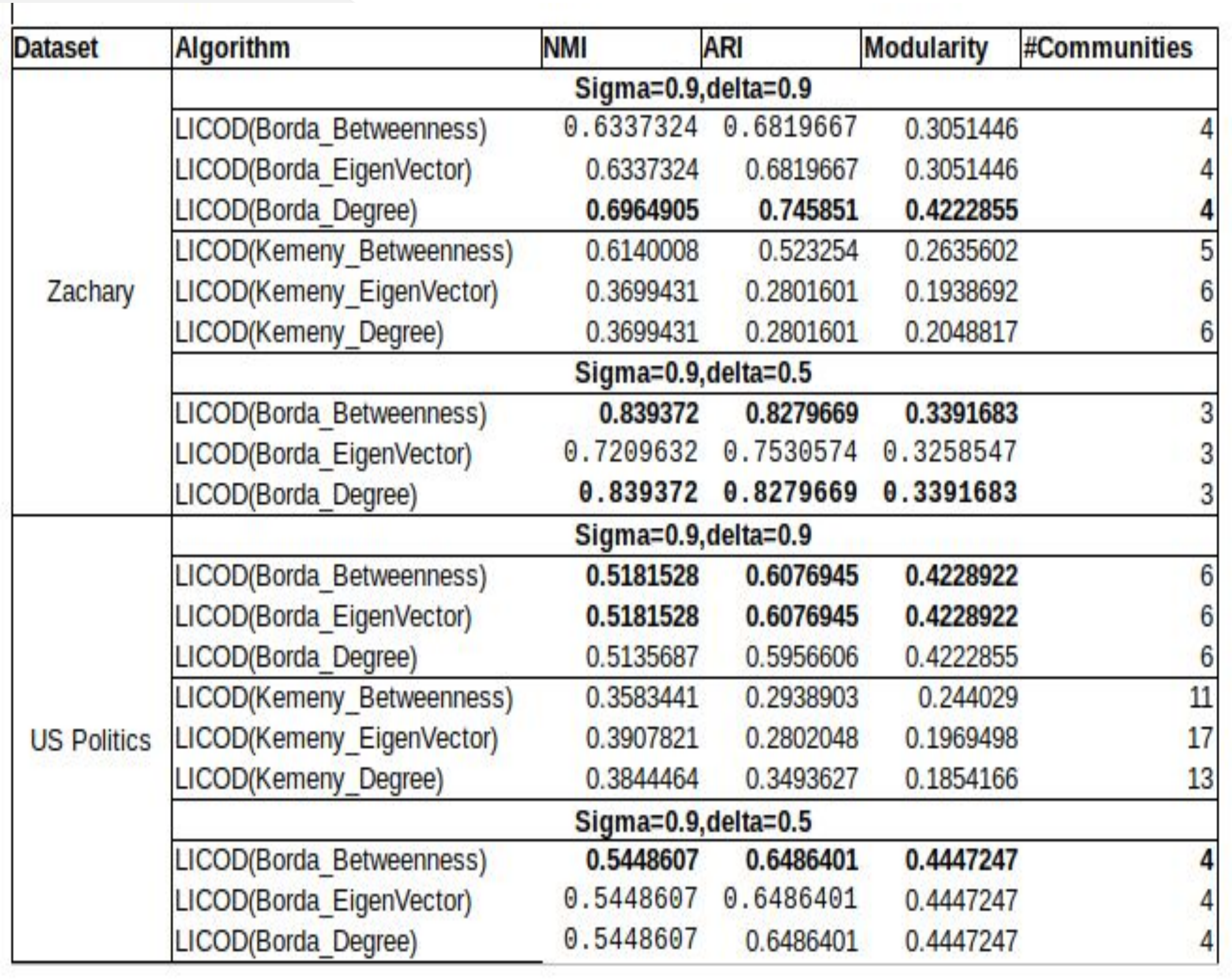
- For each network we have applied the proposed algorithm by changing the configuration parameters.
- We evaluated the impact of variations in the parameters σ,δ and Epsilon
- For each configuration, we computed the NMI, ARI and the modularity Q.
- We found out which configuration accelerate slightly the convergence for the right value to obtain.

Experimental Results and Analysis

- Variation in σ has significant impact on the results obtained.
- The best results are obtained for σ around 0.8, 0.9.
- Epsilon has negligible impact on obtained results.
- Increasing Epsilon results in diminishing the NMI and ARI.
- High value of Epsilon increases the overlapping degree of obtained communities while real communities we have here are all disjoint.
- Reducing δ decreased the number of communities obtained for certain datasets and give good values for NMI,ARI.
- Best values of δ are 0.9 and 0.5
- We observe the variation of NMI, ARI and Q, for each of the possible configurations depending on the choice of the used centrality and the voting method.
- Results show that the use of the betweenness centrality accelerate slightly the convergence for the right value to obtain.
- Borda out performs Local Kemeny in all the networks.
- Performance of each configuration differ from one network to another and it has close relation with topology and speciality of network

Experimental Results and Analysis

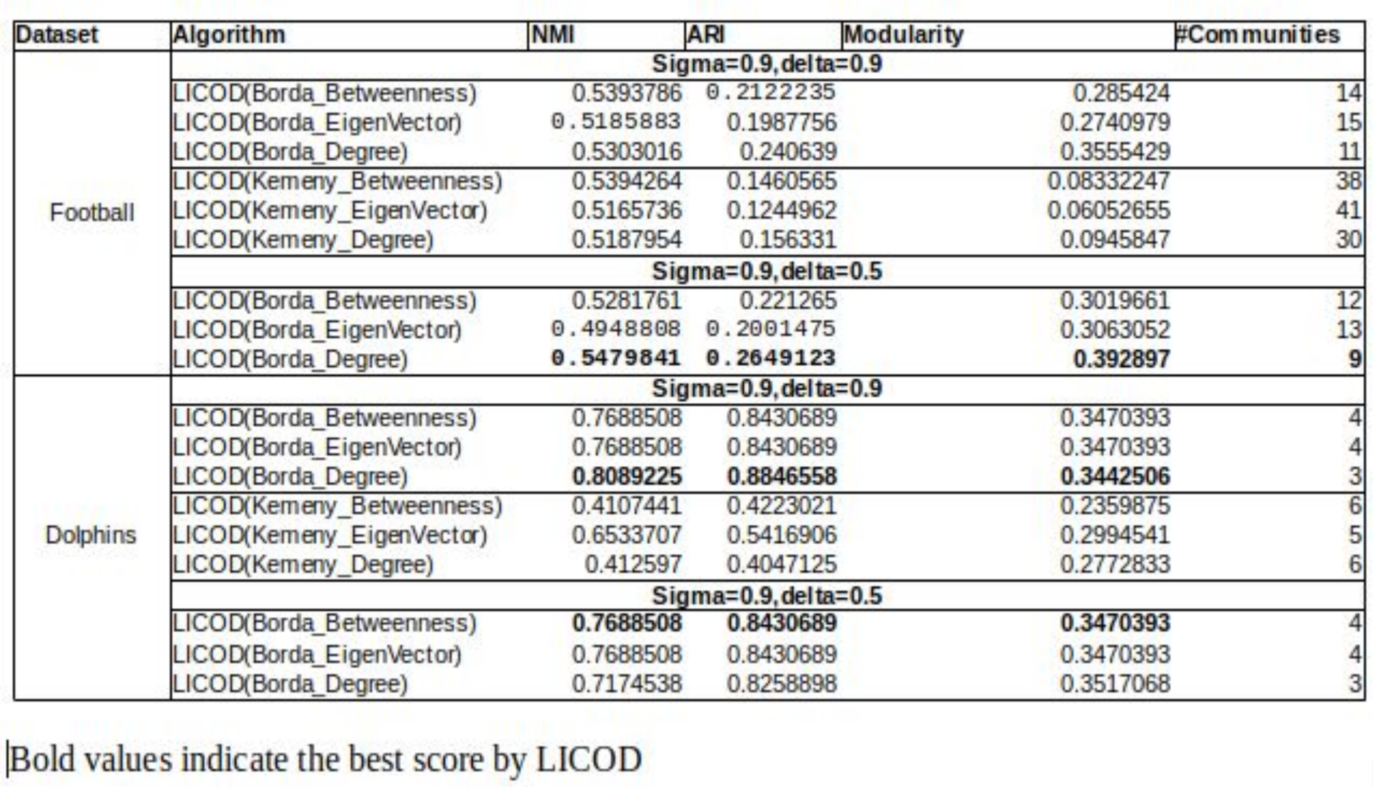
Comparison of performances of applying LICOD to different configuration parameters



Dataset	Algorithm	NMI	ARI	Modularity	#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

Experimental Results and Analysis

Comparison of performances of applying LICOD to different configuration parameters



Dataset	Algorithm	NMI	ARI	Modularity	#Communities
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

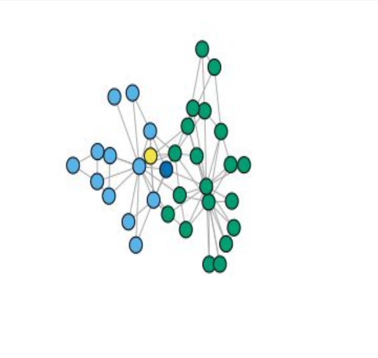
Bold values indicate the best score by LICOD

Experimental Results and Analysis

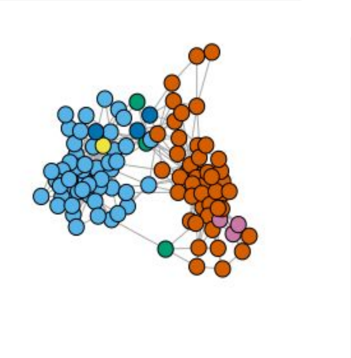
- We also compared the results of our algorithm with results obtained by well-known algorithms:
  - **Newman–Girvan algorithm**
  - **WalkTrap algorithm**
  - **Louvain algorithm**
- The configuration adopted for proposed algorithm is the following: Centrality metric is betweenness centrality, Voting method is Borda, $\sigma = 0.9$ $\delta = 0.9/0.5$, and epsilon = 0
- Results show that LICOD performs better than the other algorithms for Zachary,Dolphin and US Politics networks.
- This could be explained by the absence of leaders in other networks, which makes the communities detection task more difficult.

Experimental Results and Analysis

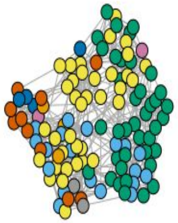
Community structure of the four selected benchmark networks obtained by LICOD (Centrality metric:betweenness centrality, Voting method:Borda, σ =0.9, δ = 0.9,and Epsilon = 0)



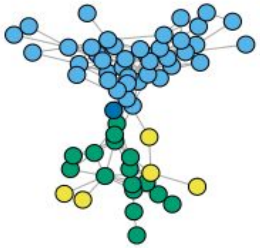
Zachary Karate Club Network



US Politics books network



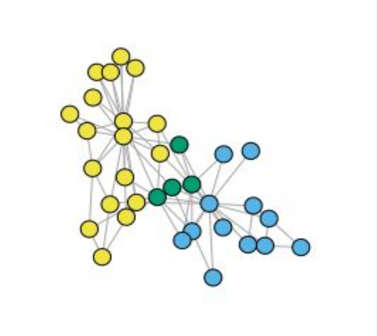
College football network



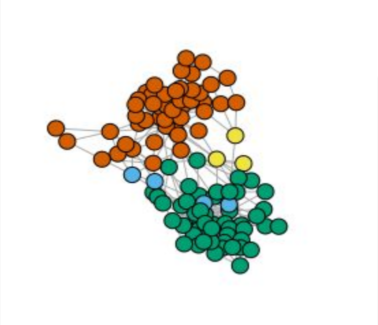
Dolphins social network

Experimental Results and Analysis

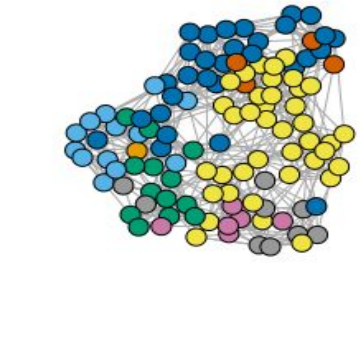
Community structure of the four selected benchmark networks obtained by LICOD (Centrality metric:betweenness centrality, Voting method:Borda, σ =0.9, δ = 0.5,and Epsilon = 0)



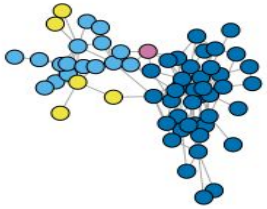
Zachary Karate Club Network



US Politics books network



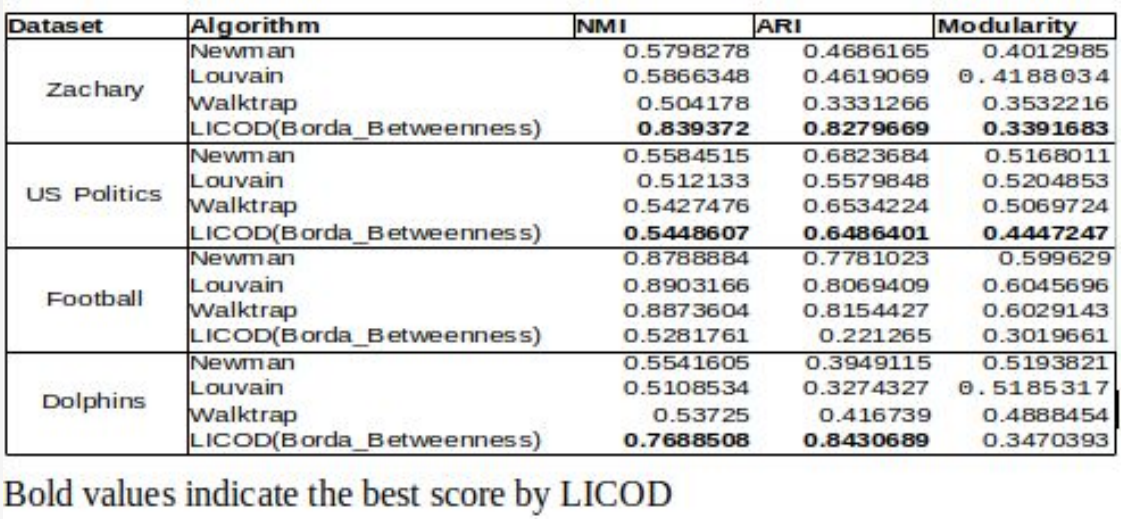
College football network



Dolphins social network

Experimental Results and Analysis

Comparison of performances of different community detection algorithms



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

Bold values indicate the best score by LICOD

- We can conclude that modularity metric does not correspond to the best decomposition into communities as measured by both NMI and ARI
- 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.

Results of Innovative Work

- Though Eigenvalue centrality is expected to give good results it doesn’t work best for the network we chose as the performance of each parameter configuration has a close relation to the specialities of the network.
- Borda out performs all the different Kemeny aggregation approaches including Local Kemeny in all the networks.
- When we tested the proposed algorithm on large scale networks it took a long time for convergence indicating that a modification is essential to handle them.

Conclusion and Future Work

- We contribute to the state of the art on community detection in complex networks by providing an efficient community detection algorithm.
- Its capacity to detect communities is demonstrated by small benchmark social networks.
- Use data clustering to apply a task-driven evaluation of community detection algorithms.
- Develop a full distributed self-stabilizing version for large scale networks.
- Adapt the approach for K-partite and for multiplex networks

Individual Contribution

- Both the team members contributed equally in the code development of **licod.R** and **main.R**
- **IdentifyCommunityLeaders.R** and **IdentifyLeaders.R** were coded by the team member Bavya Balakrishnan
- **BordaRanking.R, kemeny.R and epsilon_threshold_graph**.R were coded by team member Prajwal M P

References

- Grabowski, S., Bieniecki, W.: Tight and simple web graph compression. CoRR, abs/1006.0 (2010)
- Girvan, M., Newman, M.E.J.: Community structure in social and biological networks. PNAS 99(12), 7821–7826 (2002)
- 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)
- 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)
- 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)
- Blondel, V.D., Guillaume, J.-L., Lefebvre, E.: Fast unfolding of communities in large networks. J. Stat. Mech. Theory Exp. 2008, P10008 (2008)