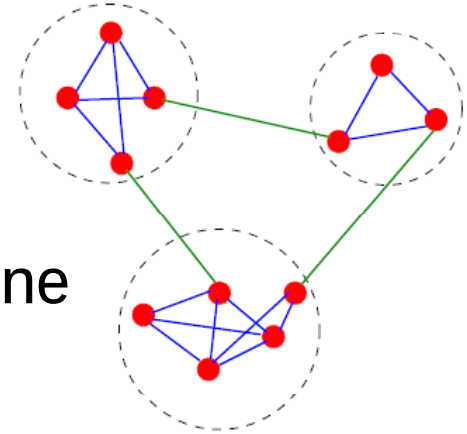
ℓ1-graph based community detection in online social networks

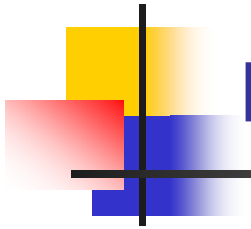
Liang Huang[1], Ruixuan Li[1], Kunmei Wen[1],
Xiwu Gu[1], Yuhua Li[1] and Zhiyong Xu[2]

[1]Huazhong University of Science and Technology
[2]Suffork University

Online social network

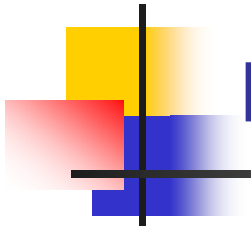
- Online social networks have attracted more and more attention in recent years.
- In online social networks, a common feature is the community structure.
- Detecting community structures in online social network is a challenging job for traditional algorithms.





Features of online social network

- The scale of online social networks are becoming very huge due to the scale of the internet.
- The relationships between people are becoming more complicated.

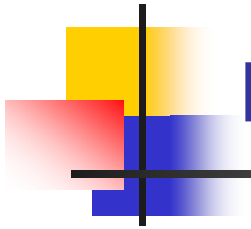


Related Work

- Newman proposed an iterative, divisive method based on the progressive removal of links with the largest betweenness.

- Santo Fortunato developed an algorithm of hierarchical clustering that consists of finding and removing the edges iteratively with the highest information centrality.

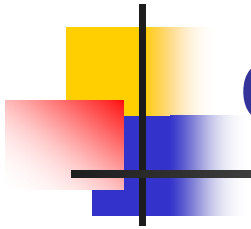
- Newman and Girvan proposed a quantitative method called modularity to identify the network communities.



Problems of current algorithms

- Uncompetitive for the large social network. The computation overhead of the current algorithms is becoming very high when handling the large-scale online social networks.

- Not consider the condition of directed social networks for normal spectral clustering algorithms.



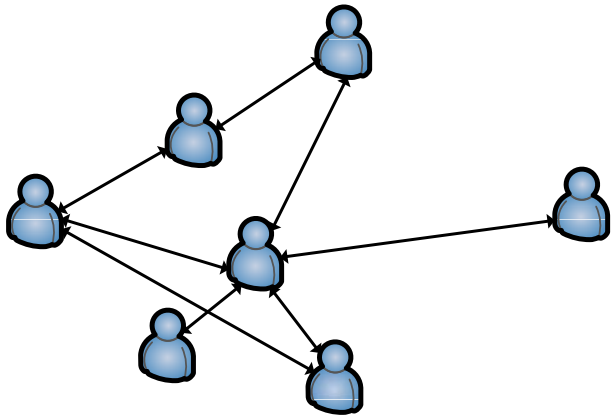
Our contributions

- We use $\ell 1$-graph to utilize the overall contextual information instead of only pairwise Euclidean distance as conventionally.

- We use the laplacian regularizer to choose relatively small sample sets of the given social network, which largely reduces the computational cost.

- We combine the $\ell 1$-graph and laplacian regularizer to detect the communities in both directed and undirected social networks.

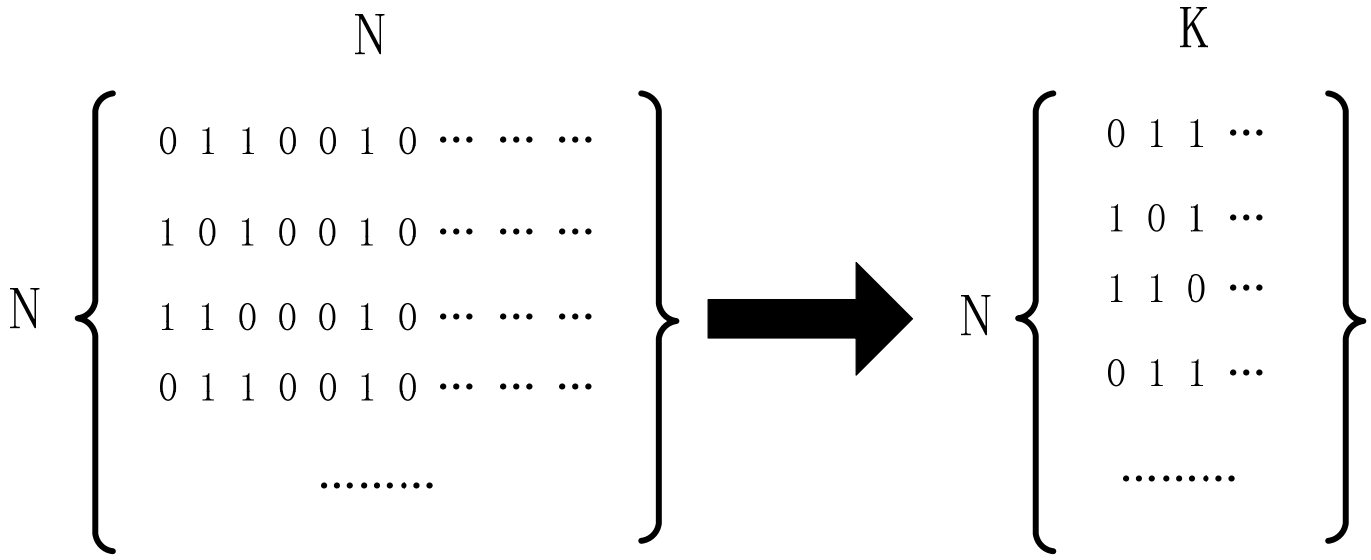
Typical social network



$$G = \langle V, E \rangle$$

$$\begin{array}{c} \\ N \end{array} \begin{array}{c} \overset{N}{\left\{ \begin{array}{l} 0\ 1\ 1\ 0\ 0\ 1\ 0 \cdots \cdots \cdots \\ 1\ 0\ 1\ 0\ 0\ 1\ 0 \cdots \cdots \cdots \\ 1\ 1\ 0\ 0\ 0\ 1\ 0 \cdots \cdots \cdots \\ 0\ 1\ 1\ 0\ 0\ 1\ 0 \cdots \cdots \cdots \\ \qquad\qquad \cdots\cdots\cdots \end{array} \right.} \end{array}$$

Spectral clustering algorithm (SCA)



Spectral clustering algorithm (SCA)

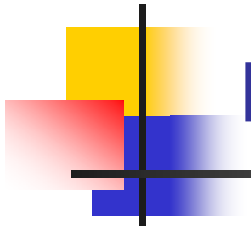
Input:

The social matrix of social graph $G = (V, E)$;

Output:

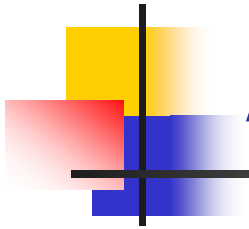
The k clusters of the social network.

1: Compute the similarity matrix of the social matrix;
2: Compute the graph laplacian matrix of the similarity matrix;
3: Compute the eigenvalues and the eigenvectors of the laplacian matrix;
4: Use the eigenvectors of the first k eigenvalues to cluster the network;
5: **return** The clusters of the given social network.



Pairwise similarity graph for SCA

- Pairwise similarity graph for SCA
  - The $\mathcal{E}$-neighborhood graph
  - k-nearest neighbor graph
  - The fully connected graph
- Shortcomings for pairwise similarity graph approaches
  - Pairwise similarity graph doesn't integrate the overall contextual information of the social networks.



ℓ1-graph for SCA

- Advantages for ℓ1-graph
  - Robustness to data noise
  - Good at represent sparsity
  - Datum-adaptive neighborhood

- Advantage for using ℓ1-graph in SCA
  - Using the ℓ1-graph, the SCA algorithm can automatically select the neighbors for each datum, and the similarity matrix is automatically derived from the calculation of these sparse representations.

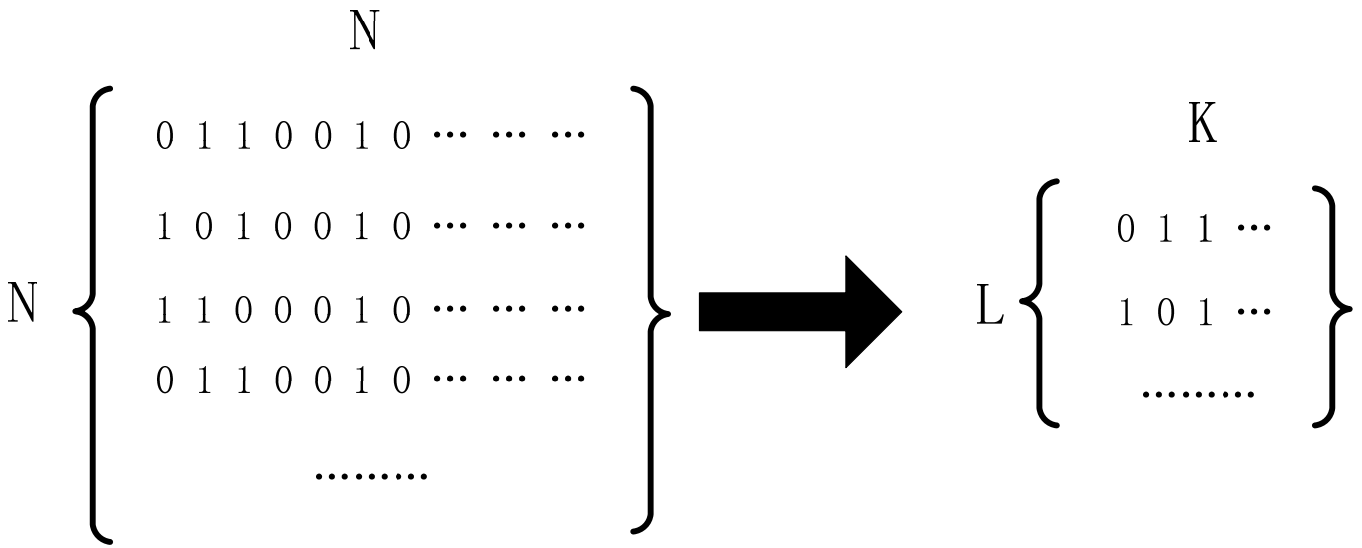
L-1 graph construction

- **Inputs:** The sample data set denoted as the matrix X= $[x_1, x_1, ...x_N]$
  - **where** $x_i \in \Re^m$
- **Robust sparse representation:**

$$\arg\min_{\alpha^i} \|\alpha^i\|, \qquad s.t. \quad x_i = B^i \alpha^i.$$

  - **Where** $B^i = [x_1, x_1, ..., x_N, I] \in \Re^{m+N-1}$
- **Graph weight setting:** $w_{ij} = \begin{cases} \alpha_j^i, & i > j, \\ 0, & i = j, \\ \alpha_{j-1}^i, & i < j. \end{cases}$

Regularized spectral clustering algorithm (RSCA)



Our algorithm – RSCA with $\ell 1$-graph

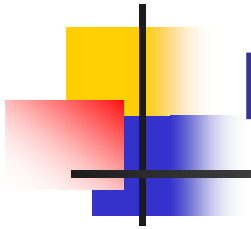
Algorithm 1 The methodology of our paper

Require:

The social matrix of social network graph $G = (V, E)$;

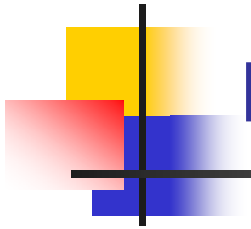
Ensure:

1: Random choose the appropriate sample set of the online social network;
2: Use the ℓ^1-graph to construct the similarity graph of the sample set;
3: Compute the graph laplacian matrix, if the social network is a directed one, use the method described in [16] to compute the directed graph laplacian matrix;
4: Compute the generalized eigenvectors and eigenvalues of the graph laplacian matrix;
5: Use the vectors of the first k eigenvalues to compute the corresponding coefficient α in (9);
6: Compute the target function values of the whole social network data correspond to the respective coefficient α;
7: Use the k vectors of the target function values to cluster the social network;
8: **return** The clusters of the given social network.



Experiments

- Implementation using Matlab.
- Data Sets:
  - Zachary
  - Dolphin network
  - Arxiv HEP-PH collaboration network
- Metric: Modularity



Modularity

- We define a quantity function of the community detection and detect the good communities through optimizing the quantity function *Q*.

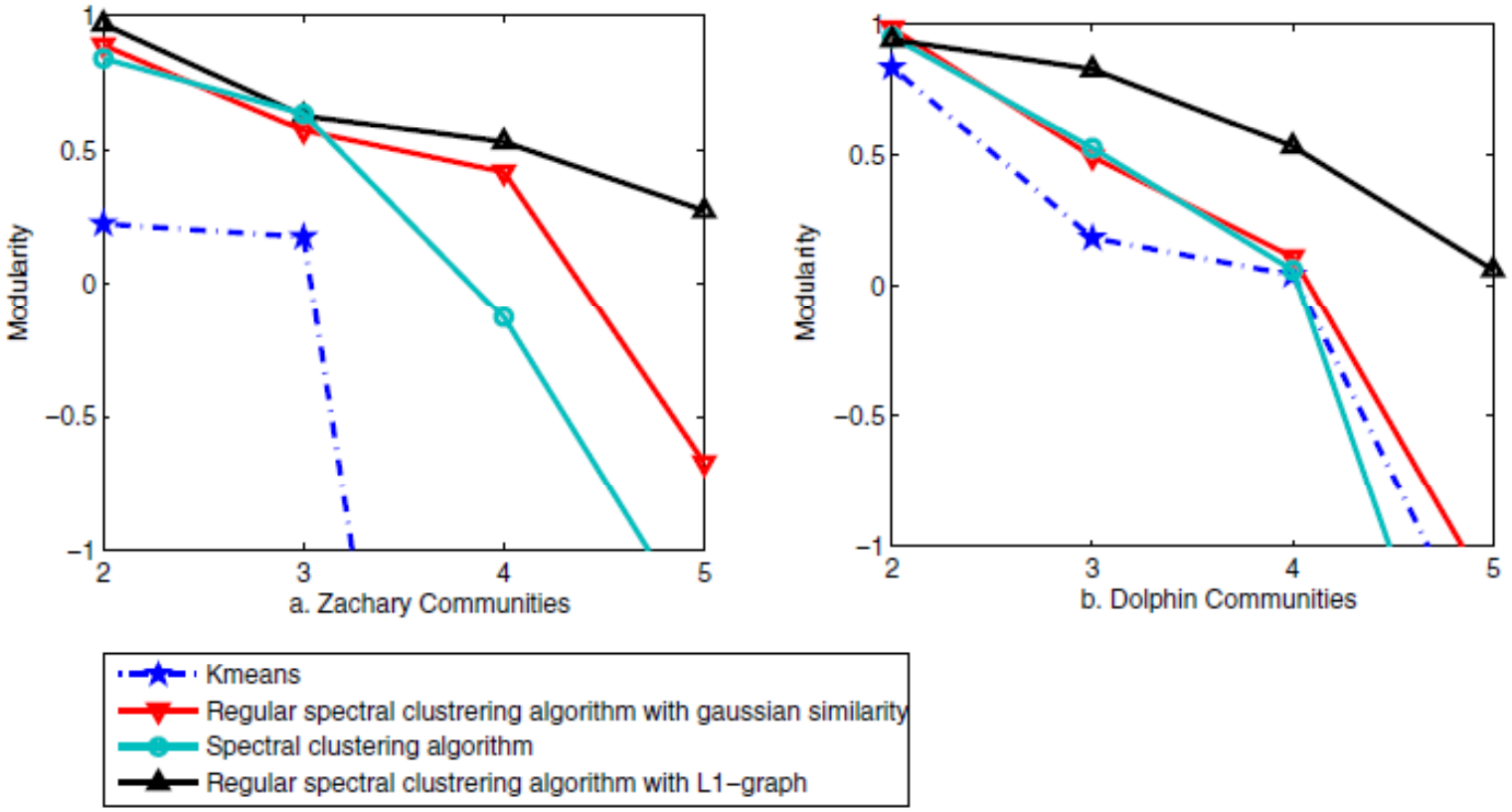
$$Q = \sum_{j} (e_{ii} - a_i^2)$$

- The more links within the communities and fewer links between communities, the more Q quantities. The maximum of Q is 1.

- We use modularity to quantify the results of our algorithm.

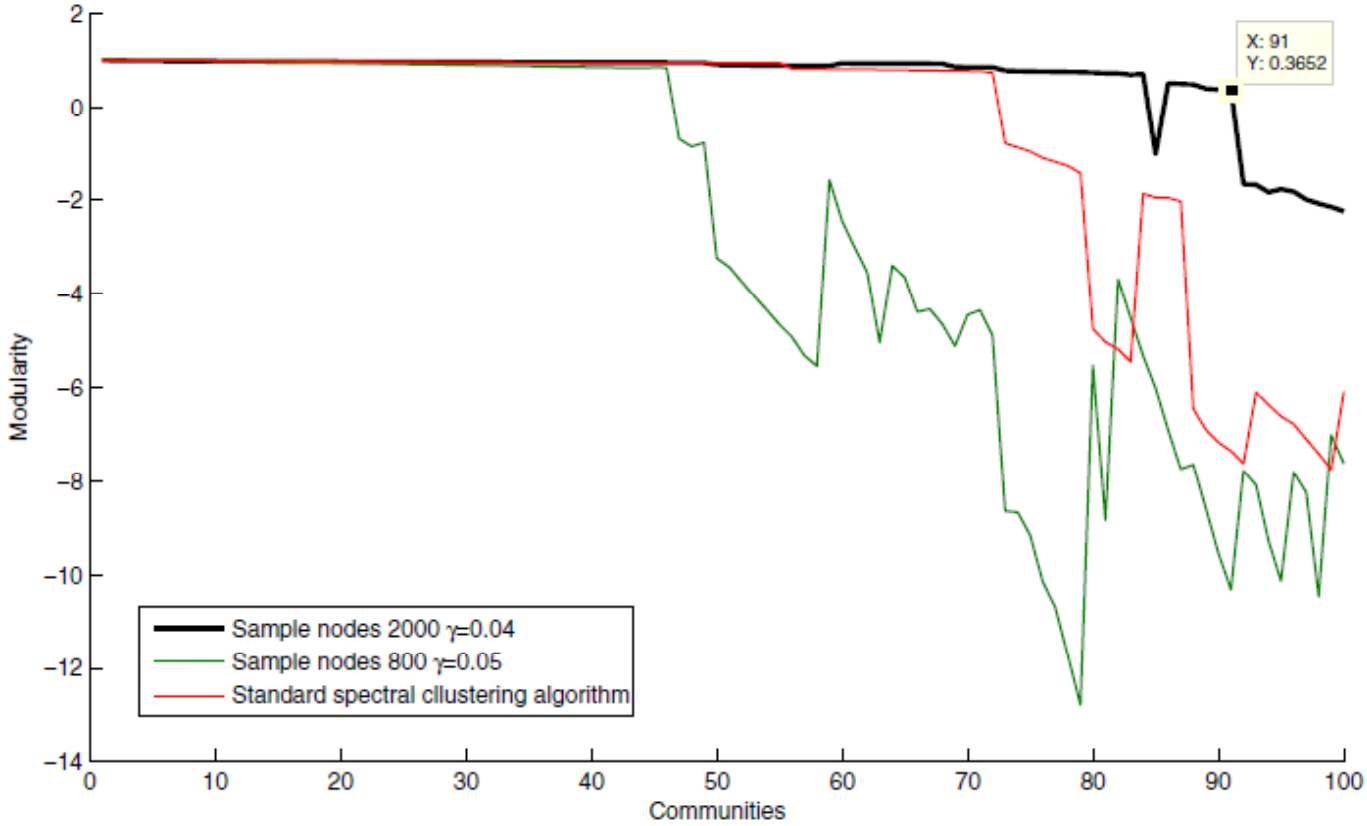
Results in two benchmark social networks (Zachary and Dolphin network)

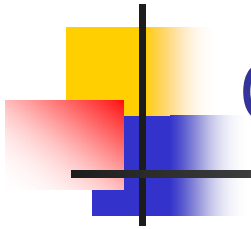
- RSCA with L1-graph and full samples has the best performance as displayed in the figure.



Result in Arxiv HEP-PH collaboration network

- Arxiv with 12,008 nodes and 237,010 edges.
- RSCA with 2000 samples has the best performance.
- SCA outperformed the RSCA with 800 samples.





Conclusion

- We proposed to integrate $L1$-graph and RSCA to detect community structures in complex online social networks in an efficient way.

- The social networks adopted in our paper only consist of tens of thousands nodes. We plan to employ some parallel and distributed computing tools, such as MapReduce, to improve the computing performance.

Thanks!