A Diffusion of Innovation-Based Closeness Measure for Network Associations

Reihaneh Rabbany, Osmar R. Zaïane

Department of Computing Science
University of Alberta
Edmonton, Canada
rabbanyk,zaiane@ualberta.com

COMMPER Workshop on Mining Communities and People Recommenders
in conjunction with the IEEE International Conference on Data Mining (ICDM)
Vancouver, Canada
December 2011
How do we define a community?

Several definitions for communities in the network

- Density of *edges* within is greater than between
- Community membership based on a notion of *similarity*

Proximity measures based on feature vectors
  e.g. Euclidean distance, Mahalanobis distance

**iCloseness**

New closeness measure appropriate for mining communities

- Encapsulates the notion of membership in a community
- Reinforced by observations made on community dynamics in social networks with regard to the probability of joining a group based on the concept of Diffusion of Innovation *(Backstrom et al., 2006)*
What is the basic idea of iCloseness?

The likelihood of joining a community in social networks depends upon the number of pre-existing connections with group members and the density of edges between these members and other members in the group. I.e. if I am faced with two groups in which I already have friends and if I need to join one of these groups, I could choose either one but, there is a higher probability to join the group in which I have more friends. In addition if I am faced with two groups in which I have the same number of friends, there would be a higher probability that I would join the group in which the connectivity of my friends with the group is stronger.
Outline

- Introduction
- Related Works
  - Graph partitioning
  - Clustering
- Method
  - iCloseness
  - iTop Leaders
- Experiments
  - Setups
  - Comparing Measures
  - Comparing Algorithms
- Conclusions
- References
Addressing similar question as graph partitioning

- Spectral clustering method e.g. (Ng et al., 2001)
  the sizes of the groups need to be fixed
- Cut minimization methods e.g. *ratio cut* (Chan et al., 1993),
  *normalized cut* (Shi & Malik, 2000), and the *min-max cut* (Ding et al., 2001) in favour of divisions into equal-sized parts

Community structure detection assumes that the networks divide **naturally** into some partitions and there is no reason that these partitions should be of the same size
Hierarchical clustering as the standard method

Hierarchical clustering: discovers natural divisions of social networks into groups, based on

- a metric measuring the similarity/closeness between vertices
- a quality function measuring the quality of a particular division (Chen et al., 2009; Girvan & Newman, 2002; Guimera et al., 2004; Newman & Girvan, 2004; Nicosia et al., 2008)

**Modularity Q** (Newman & Girvan, 2004; Clauset et al., 2004), is a well-known example of such quality function in finding communities and serves as the basis of many other later proposed metrics.

The iCloseness measure is based on the connection between theoretical models of diffusion in social networks to the community membership investigated by (Backstrom et al., 2006)
Definition of iCloseness

Defined based on theory of *Diffusion of Innovations* – a theory of how, why, and at what rate new ideas and technology spread through cultures. Specifically for the case of information networks, if we consider the act of joining a community as a behaviour that spreads within a network the idea of Diffusion of Innovations very naturally extends to community mining. Socially, there is indeed advantage in joining a group that already includes friends that know each other and who are connected.

(Backstrom *et al.*, 2006)
Definition of iCloseness

Measures closeness between two nodes and indicates how much these nodes tend to belong to the same community $\Rightarrow$ calculated based on their common neighbours

- Intersection of neighbourhoods
- Density of common neighbours
- Expanding neighbourhoods

Node $n$ should be assigned to community of leader $L_1$
Neighbours Scoring

\( \mathcal{N}(v) \): neighbourhood of node \( v \)

i.e. \( u \in \mathcal{N}(v) \) iff there exists a path of length at most \( \delta \) to that node from \( v \)

- **neighbours scoring**: based on the depth of their neighbourhood and based on how dense they are connected

\[
n s_1(u, v) = \begin{cases} 
 1 & \text{if } e(u, v) \in E \\
 0 & \text{otherwise}
\end{cases}
\]

\[
E_1(v) = \{ e(u, v) \in E \}
\]

\[
n s_l(u, v) = n s_{l-1}(u, v) + \sum_{e(u, m) \in E - E_{l-1}(v)} n s_{l-1}(m, v) \times \frac{e(u, m)}{\sum_i e(i, m)}
\]

\[
E_l(v) = E_{l-1}(v) \cup \{ e(i, j) \in E \mid \exists e(j, k) \in E_{l-1}(v) \}.
\]
An example of scored neighbourhood

Closer nodes to node 9 have higher neighbouring scores; neighbours that are more densely connected to 9, also obtain higher scores
Computing iCloseness

The iCloseness of node $v_1$ and $v_2$, is obtained as:

$$iCloseness(v_1, v_2) = \sum_{u \in \mathcal{N}(v_1) \cap \mathcal{N}(v_2)} ns(u, v_1) \times ns(u, v_2)$$

The neighbourhood threshold ($\delta$) determines the maximum neighbourhood level which should be tuned based on the application e.g. a small number (e.g. 3) for social networks, because of the six-degree separation theory.
Example revised: with iCloseness values

Scored neighbourhoods

Marked by iCloseness values

The same example but marked with iCloseness of all nodes to node 9
Applying iCloseness in community detection

Top Leaders first introduced in (Rabbany Khorasgani et al., 2010)
Inspired by the well-known k-medoids clustering algorithm

Iteratively

- elects $k$ representative nodes as leaders
- associats followers to one of these leaders to form communities based on the relations/links between nodes

We adopted the original Top Leaders algorithm to use iCloseness when determining the community memberships of nodes
iTop Leaders algorithm

initialize k leaders

repeat
  {finding communities}
  for all Node $v \in G$ do
    if $v \not\in$ leaders then
      find community of $v$
    end if
  end for
  {updating leaders}
  for all $\ell \in$ leaders do
    $\ell \leftarrow \arg \max_{v \in \text{Community}(\ell)} \text{Centrality}(v)$
  end for
until there is no change in the leaders

Leader
  the most central member in its community

Community
  a leader and the follower nodes associated with it
Association of followers to the *iClosest* leader

\[
\text{depth} \leftarrow 1
\]
\[
\text{CanList} \leftarrow \text{leaders} \cap \mathbb{N}(\mathbb{N}(v))
\]
\[
\text{CanList} \leftarrow \text{arg max}_{\ell \in \text{CandList}} \text{iCloseness}(v, \ell)
\]
\[
\text{if } |\text{CanList}| = 0 \text{ then } \{\text{No candidate leader}\}
\]
\[
\text{if Centrality}(v) < \epsilon \text{ then } \{\text{Noise}\}
\]
\[
\text{associate } v \text{ as an outlier}
\]
\[
\text{else } \{\text{Powerful but free}\}
\]
\[
\text{associate } v \text{ as a hub}
\]
\[
\text{end if}
\]
\[
\text{else if } |\text{CanList}| > 1 \text{ then } \{\text{Many candidates}\}
\]
\[
\text{associate } v \text{ as a hub}
\]
\[
\text{else } \{\text{Only one candidate leader in CanList}\}
\]
\[
\text{associate } v \text{ to the only leader in CanList}
\]
\[
\text{end if}
\]

- the *iCloseness*
- the view
- speed
- *k*
Updating leaders

The election of the node with the highest centrality in each community

\[ \ell \leftarrow \arg \max_{v \in \text{Community}(\ell)} \text{Centrality}(v) \]

The centrality of nodes in a community measures the relative importance/popularity of a node within that group
Leaders are central nodes in their community ⇒ the $k$ most central nodes that none of them belong to the same community

Starts from the most central node, and adds the next central one to the current set of leaders only if it is not too $iClose$ to any of the current leaders

A detailed comparison of different initialization methods for the Top Leaders is presented in (Rabbany Khorasgani et al., 2010)
Outline

- Introduction
- Related Works
- Method
- Experiments
  - Setups
  - Comparing Measures
    - Comparing on real world benchmarks
    - Comparing on synthesized benchmarks
  - Comparing Algorithms
    - Comparing on real world benchmarks
    - Results on synthesized benchmarks
    - Comparing on large scale real-world data set
    - Parameters
    - Complexity
- Conclusions
- References
Evaluating community mining methods

Well-known (small) real world data-sets

Comparing using a measures of agreement
- **Adjusted Rand Index (ARI)** (Santos & Embrechts, 2009)
  \[ -1(\text{no agreement at all}) \ldots 1(\text{full agreement}) \]
- **Normalized Mutual Information (NMI)** (Lancichinetti & Fortunato, 2009)
  \[ 0(\text{partitions are independent}) \ldots 1(\text{partitions are identical}) \]

Karate and WKarate Club (weighted) (Zachary, 1977), Sawmill Strike (Pajek, n.d.),
NCAA Football Bowl Subdivision (Xu et al., 2007), and Politician Books (Krebs, n.d.)

Synthesized with characteristics similar to real-world networks

LFR benchmarks (Lancichinetti et al., 2008): Power-law distribution over degrees and community sizes & Mixing parameter $\mu$: each node shares a fraction of its edges with the nodes of other communities

Large real networks with no ground-truth

Co-purchasing network of Amazon.com (Clauset et al., 2004)
815, 223 nodes and 3,426, 127 edges
## Results on real world benchmarks

<table>
<thead>
<tr>
<th>dataset</th>
<th>measure</th>
<th>ARI</th>
<th>NMI</th>
</tr>
</thead>
<tbody>
<tr>
<td>Karate</td>
<td>Shortest Path</td>
<td>.669</td>
<td>.655</td>
</tr>
<tr>
<td>(2 groups, 34</td>
<td>Adjacency Relation</td>
<td>.771</td>
<td>.732</td>
</tr>
<tr>
<td>nodes, 78 edges)</td>
<td>Neighbour Overlap</td>
<td>.446</td>
<td>.383</td>
</tr>
<tr>
<td></td>
<td>Pearson Correlation</td>
<td>.328</td>
<td>.324</td>
</tr>
<tr>
<td></td>
<td><strong>iCloseness</strong></td>
<td>1.</td>
<td>1.</td>
</tr>
<tr>
<td>Strike</td>
<td>Shortest Path</td>
<td>.935</td>
<td>.926</td>
</tr>
<tr>
<td>(3 groups, 24</td>
<td>Adjacency Relation</td>
<td>.903</td>
<td>.834</td>
</tr>
<tr>
<td>nodes, 38 edges)</td>
<td>Neighbour Overlap</td>
<td>.819</td>
<td>.763</td>
</tr>
<tr>
<td></td>
<td>Pearson Correlation</td>
<td>.109</td>
<td>.307</td>
</tr>
<tr>
<td></td>
<td><strong>iCloseness</strong></td>
<td>1.</td>
<td>1.</td>
</tr>
<tr>
<td>PolBooks</td>
<td>Shortest Path</td>
<td>.647</td>
<td>.542</td>
</tr>
<tr>
<td>(2 groups, 105</td>
<td>Adjacency Relation</td>
<td>.630</td>
<td>.573</td>
</tr>
<tr>
<td>nodes, 441 edges)</td>
<td>Neighbour Overlap</td>
<td>.687</td>
<td>.585</td>
</tr>
<tr>
<td></td>
<td>Pearson Correlation</td>
<td>.053</td>
<td>.157</td>
</tr>
<tr>
<td></td>
<td><strong>iCloseness</strong></td>
<td>.769</td>
<td>.696</td>
</tr>
<tr>
<td>Football</td>
<td>Shortest Path</td>
<td>.689</td>
<td>.559</td>
</tr>
<tr>
<td>(11 groups,</td>
<td>Adjacency Relation</td>
<td>.431</td>
<td>.753</td>
</tr>
<tr>
<td>180 nodes,</td>
<td>Neighbour Overlap</td>
<td>.970</td>
<td>.948</td>
</tr>
<tr>
<td>787 edges)</td>
<td>Pearson Correlation</td>
<td>.082</td>
<td>.007</td>
</tr>
<tr>
<td></td>
<td><strong>iCloseness</strong></td>
<td>.996</td>
<td>0.989</td>
</tr>
</tbody>
</table>
Comparison of different measures on LFR synthesized benchmarks. The horizontal axis represent datasets with different mixing parameter $\mu$, and the vertical axis is the accuracy. Different curves stand for different measures.

$|V| = 1000$, $\text{degree}_{\text{avg}} = 20$, $\text{degree}_{\text{max}} = 50$, $\mu = 0 : 1 : 1$, $|C|_{\text{min}} = 20$, $|C|_{\text{max}} = 100$
Comparing iTop Leaders with other algorithms

iTop Leaders equipped with iCloseness v.s. three well-known community mining approaches

1. **FastModularity** (Clauset et al., 2004)
   Community: a particular partitioning with maximum quality compared to random

2. **CFinder** (Palla et al., 2005)
   Community: union of adjacent cliques

3. **SCAN** (Xu et al., 2007)
   Community: nodes that are structurally reachable
Results on real world benchmarks

a) Karate, b) Strike, c) Politician Books and d) Football

- **the red bar**: accuracy of results obtained by opponent method
- **the blue bar**: result of iTop Leaders seeded with the same $k$
- **the green bar**: the ideal when iTop Leaders is seeded with correct $k$
Comparing Algorithms

Results on synthesized benchmarks

The first three plots compare ARI of iTop Leaders (iTL) and other contenders as a function of mixing parameter $\mu$. The last one shows the results of iTop Leaders given the correct $k$. 

- **SCAN**
- **CFinder**
- **FastModularity**
- **GroundTruth**
Results on large scale real-world data set

Amazon network

- CFinder: did not terminate successfully
- SCAN: did not terminate successfully
- FastModularity: x10 slower, $Q = 0.77$
- iTop Leaders with $k_{FM}$, $Q = 0.45$

When ground truth is not available, modularity ($Q$) is typically used to assess the quality of discovered communities

- $0.3 \geq$ shows a significantly good partitioning (Clauset et al., 2004)
- higher modularity does not ensure a higher accuracy
- our algorithm detected 89865 hubs
  are not member of any specific community and this decreases the modularity significantly
**Parameters**

**iCloseness**

- the **neighbourhood threshold** ($\delta$) based on the application e.g. 3 for social networks

**Top Leaders**

- the **number of desired communities** ($k$)
  
  *Top leaders* always improves the results with same $k$

  While $k$ is mostly not correct or even close e.g. in the synthesized benchmarks with $33\pm5$ ⇒
  
  FastModularity: $12\pm6$, CFinder: $1182\pm464$, and Scan: $299\pm127$

- the **outlier threshold** ($\gamma$), and the **hub threshold** ($\epsilon$)
  
  Control the characteristics of the final output

  If both zero ⇒ no outliers and disjoint clusters
Comparing Algorithms

Running time of iTop Leaders v.s. other methods

The running time of iTop Leaders, FastModularity, CFinder and SCAN
Conclusions

- Intersection Closeness to assess the proximity of a node to a community representative
- Based on the theory of diffusion of innovation, which states that the probability of joining a group depends on the number of existing friends in the group and their connectedness
- The experimental results of Top Leaders algorithm equipped with $iCloseness$ show high accuracy and effectiveness in both real and synthesized networks, compared against commonly used closeness/distance measures as well as the state-of-the-art community mining methods

- Questions?
Backstrom, Lars, Huttenlocher, Dan, Kleinberg, Jon, & Lan, Xiangyang. 2006. 
Group formation in large social networks: membership, growth, and evolution. 
In: ACM SIGKDD.

Spectral k-way ratio-cut partitioning and clustering. 

Detecting Communities in Social Networks using Max-Min Modularity. 
In: SIAM Data Mining Conference.

Finding community structure in very large networks. 

A Min-max Cut Algorithm for Graph Partitioning and Data Clustering. 

Community structure in social and biological networks. 
Proceedings of the National Academy of Sciences of the USA.

Modularity from fluctuations in random graphs and complex networks. 

Krebs, Valdis. 
Books about US politics. 
Lancichinetti, Andrea, & Fortunato, Santo. 2009.
Community detection algorithms: A comparative analysis.

Benchmark graphs for testing community detection algorithms.
*Physical Review E*, 78, 046110.

Finding and evaluating community structure in networks.

On spectral clustering: Analysis and an algorithm.
*Pages* 849–856 of: *NIPS*.

Extending modularity definition for directed graphs with overlapping communities.

Pajek.

http://vlado.fmf.uni-lj.si/pub/networks/pajek/.

Uncovering the overlapping community structure of complex networks in nature and society.

Top leaders Community Detection Approach in Information Networks.
*In: 4th SNA-KDD Workshop on Social Network Mining and Analysis*.

Santos, Jorge M., & Embrechts, Mark. 2009.
On the Use of the Adjusted Rand Index as a Metric for Evaluating Supervised Classification.
*Pages* 175–184 of: *ICANN* (2).
Normalized cuts and image segmentation.  
*IEEE. Trans. on Pattern Analysis and Machine Intelligence.*

*Pages 824–833 of: ACM SIGKDD.*

An information flow model for conflict and fission in small groups.  
*Journal of Anthropological Research, 33, 452–473.*