Top Leaders Community Detection Approach in Information Networks

Reihaneh Rabbany Khorasgani, Jiyang Chen, Osmar R. Zaïane

Department of Computing Science
University of Alberta
Edmonton, Alberta, Canada T6G 2E8
{rabbanyk, jiyang, zaiane}@cs.ualberta.ca

4th SNA-KDD Workshop on Social Network Mining and Analysis – Washington DC, July 2010
What is Community Structure?

- **Community structure** denotes the existence of densely connected groups of nodes, with only sparser connections between groups.
- Many social networks share the property of a community structure, e.g., WWW, tele-communication networks, academic collaboration networks, friendship networks, etc.

Many similarities with data **Clustering**
Outline

● Introduction
● Some related work
  ○ Graph Partitioning, Modularity Q and others
● Top Leaders
  ○ Problem Definition
  ○ Associating nodes to Leaders
  ○ Initialization of the leaders
  ○ Experiments
● Conclusions
Introduction

- A new Approach for Finding Communities in the Networks
  - Densely connected nodes with sparse connections outside the group

- Community: set of followers congregating around a potential leader

- Algorithm: similar in spirit to k-means
  - starts by identifying promising leaders in the network
  - iteratively until convergence
    - assembles followers to their closest leaders to form communities
    - finds new leaders in each group around which to gather followers again
Related Work

- Graph partitioning and spectral clustering approaches
  - dividing the network into groups with (roughly) equal size, while minimizing the number of edges that run between vertices in different groups

- Q-modularity by Newman 2004
  - measure of the quality of a particular division of a network

- CFinder by Palla et al. 2005
  - community: union of complete subgraphs of size $k$
  - $k$ between 3 and 5: very effective on real networks

- SCAN by Xu et al. 2007
  - nodes that are structurally reachable from each other are grouped together in the same community
Problem Definition

- Finding leaders and their followers in the network to form the communities
- Leader: The central/influential node
- Community: Set of followers surrounding a leader
- Assigning followers to closest leader based on the intersection of their neighborhood
Top Leaders Approach

A leader is the most central member in a community

Algorithm 1 Top Leaders algorithm

Input: A social network G, and k the number of desired communities

initialize k leaders
repeat
  {finding communities}
  for all Node n ∈ G do
    if n ∉ leaders then
      associate n to a leader {Algorithm 2}
    end if
  end for
  {updating leaders}
  for all l ∈ leaders do
    l ← arg max_{n ∈ Community(l)} Centrality(n)
  end for
until there is no change in the leaders
Associating Nodes to Leaders

Algorithm 2 Associate \( n \) to its leader

**Input:** Social network \( G \), node \( n \), set of \( k \) leaders

\[
\text{depth} \leftarrow 1 \\
\text{CanList} \leftarrow \text{leaders} \\
\text{repeat} \\
\quad \text{CanList} \leftarrow \underset{c \in \text{CanList}}{\text{arg max}} |N(n_1, d) \cap N(n_2, d)| \\
\quad \text{depth} \leftarrow \text{depth} + 1 \\
\quad \text{until} \ |\text{CanList}| \leq 1 \lor \text{depth} > \delta \\
\text{if} \ |\text{CanList}| = 0 \ \text{then} \ \{\text{No candidate leader}\} \\
\text{associate } n \text{ as an outlier} \\
\text{else if} \ |\text{CanList}| > 1 \ \text{then} \ \{\text{Many candidates}\} \\
\text{associate } n \text{ as a hub} \\
\text{else} \ \{\text{Only one candidate leader in CanList}\} \\
\text{associate } n \text{ to CanList} \\
\text{end if}
\]

Community membership of the nodes is association of followers to nearby leaders.
Top Leaders Approach

A leader is the most central member in a community

Algorithm 1: Top Leaders algorithm

Input: A social network $G$, and $k$ the number of desired communities

1. Initialize $k$ leaders
2. Repeat
   - Finding communities
   - For all node $n \in G$ do
     - If $n \notin$ leaders then
       - Associate $n$ to a leader \{Algorithm 2\}
     - End if
   - End for
   - Updating leaders
   - For all $l \in$ leaders do
     - $l \leftarrow \arg \max_{n \in Community(l)} Centrality(n)$
   - End for
3. Until there is no change in the leaders
Initialization Methods

Wrong leaders may get stuck in a bad local optimum

- **Naïve Initialization**
  - random selection of k nodes from the network

- **Top Global Leaders**
  - k most central nodes in the network

- **Top Leaders & not Direct Neighbour**
  - the k most central nodes that are not directly connected to each other
  - avoid choosing two correct leaders that are directly connected but truly in different communities

- **Top leaders & Few Neighbours in Common**
  - based on intersections, similar to followers association
Experiments and Datasets

Competitors
● three of other well-known community detection methods

Datasets
● Karate-Club dataset
  34 nodes in 2 communities

● Sawmill Strike dataset
  24 nodes in 3 communities

● Football dataset
  180 nodes in 11 communities
Evaluation Metrics

● Comparing with Ground Truth
  ○ Purity
    ■ the number of correctly assigned nodes divided by the total number of nodes. 0 (no agreement at all) to 1 (full agreement).
  ○ Adjusted Rand Index (ARI)
    ■ penalizes false negatives and false positives. -1 (no agreement at all) and 1 (full agreement), 0 (no better than random)

● Modularity
  ○ how well the edges fall within the detected communities compared to a randomized network. 0 (no different than a randomized network), > 0.3 (good partition)
Comparing Initialization Methods

- Naïve
- Top Global Leaders (TGL)
- Top Leaders & not Direct Neighbour (TL&NDN)
- Top Leaders & Few Neighbours in Common (TL&FNiC)

<table>
<thead>
<tr>
<th>method</th>
<th>dataset</th>
<th>ARI</th>
<th>purity</th>
<th>Q</th>
</tr>
</thead>
<tbody>
<tr>
<td>Naïve</td>
<td>Karate</td>
<td>.80±.33</td>
<td>.90±.20</td>
<td>.28±.13</td>
</tr>
<tr>
<td></td>
<td>Strike</td>
<td>.59±.25</td>
<td>.81±.13</td>
<td>.41±.12</td>
</tr>
<tr>
<td></td>
<td>Football</td>
<td>.39±.12</td>
<td>.66±.08</td>
<td>.27±.07</td>
</tr>
<tr>
<td>TGL</td>
<td>Karate</td>
<td>1.0</td>
<td>1.0</td>
<td>.37</td>
</tr>
<tr>
<td></td>
<td>Strike</td>
<td>1.0</td>
<td>1.0</td>
<td>.54</td>
</tr>
<tr>
<td></td>
<td>Football</td>
<td>.83</td>
<td>.88</td>
<td>.43</td>
</tr>
<tr>
<td>TL&amp;NDN</td>
<td>Karate</td>
<td>1.0</td>
<td>1.0</td>
<td>.37</td>
</tr>
<tr>
<td></td>
<td>Strike</td>
<td>1.0</td>
<td>1.0</td>
<td>.54</td>
</tr>
<tr>
<td></td>
<td>Football</td>
<td>.78</td>
<td>.88</td>
<td>.42</td>
</tr>
<tr>
<td>TL&amp;FNiC</td>
<td>Karate</td>
<td>1.0</td>
<td>1.0</td>
<td>.37</td>
</tr>
<tr>
<td></td>
<td>Strike</td>
<td>1.0</td>
<td>1.0</td>
<td>.54</td>
</tr>
<tr>
<td></td>
<td>Football</td>
<td>.98</td>
<td>.97</td>
<td>.51</td>
</tr>
</tbody>
</table>
Visualized Results
Karate Club
Visualized Results

Strike
Visualized Results
Football, Naïve Initialization

ARI: 0.8±0.3
Visualized Results
Football, Top Global Leaders

ARI: 0.83
Visualized Results
Football, TL & Not Direct Neighbour

ARI: 0.78
Visualized Results
Football, TL & Few Neighbour in Common

ARI: 0.98
Comparing with other approaches

- Given the correct initial k, TopLeaders always provides the best result.
- The other methods do not always find the correct k but even when seeded to Top Leaders, our approach improved the quality of the found communities based on ARI.
Conclusion

- A novel algorithm to mine communities, which assigns nodes to leaders of communities and selects the leaders of communities iteratively.
- Effective in discovering communities and also in identifying outliers in a network.
- Requires $k$, the number of desired communities as input. However, it is possible to obtain $k$ after running other contenders and provide the number of discovered communities to our algorithm;
  - Our experimental results showed that communities obtained in this way are more accurate than the original discovered communities even if the used method detected wrong number of communities.

Questions?