Community Detection

Mauro Sozio

Telecom ParisTech

December 13, 2016

Community Detection from Seed Sets

We are interested in accessing or studying a group of people in a social network (algorithmists, data scientists, hikers, ...) but we know only a few users in the group. We wish to expand this group.

Problem: Given a graph G, a set S of *seed* nodes, an integer $k > 0$, find k additional nodes belonging to the "same community" of S.

Community Detection from Seed Sets

Example: Study on *secularists* vs. *islamists* on Twitter [2].

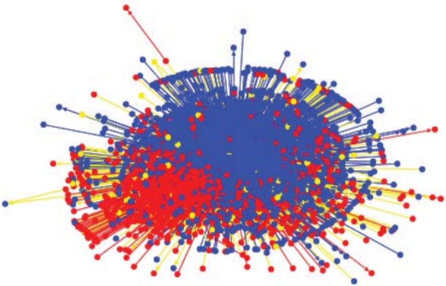

Figure: Retweet network: red nodes indicate islamists, blue nodes indicate secularists. Communities are found starting by a few known islamists/secularists.

Algorithms

Several algorithms based on:

- *local modularity* [3] (different than the one we saw): add the node increasing modularity the most.
- conductance [4]: add the node decreasing conductance the most.
- PageRank . . .

PageRank with Restart: Matrix

Let $G = (V, E)$ (web graph) be a directed graph, with $V = \{v_1, \ldots, v_n\}$. Let $\delta_{\mathrm{in}}(v)$ be the in-degree of v, i.e. $\delta_{in}(v) = |\{u : (u, v) \in E\}|$, while let $\delta_{\mathrm{out}}(v)$ be its out-degree, i.e. $\delta_{out}(v) = |\{u : (v, u) \in E\}|$.

Let M_G (M for short) be a $n \times n$ matrix with entries in $[0, 1]$ as follows:

$$M_{ij} = \begin{cases} \frac{1}{\delta_{\mathrm{out}}(v_j)} & if (v_j, v_i) \in E \\ 0 & if (v_j, v_i) \notin E \end{cases}, \quad \forall i, j \in [1, n].$$

PageRank with Restart: Matrix

Let $S \subseteq V$ be the *seed* nodes, $\beta \in (0,1)$ (probability to jump). Let $R_{G,S}$ (R for short), be a $n \times n$ matrix with entries in $[0,1]$ defined as follows:

$$R_{ij} = \begin{cases} \frac{1}{|S|} & \text{if } v_i \in S \\ 0 & \text{if } v_i \notin S \end{cases}, \quad \forall i,j \in [1,n].$$

The PageRank matrix A is then: $A_{ij} = \beta M_{ij} + (1-\beta)R_{ij}$, $i,j \in [1,n]$.

Fact: The Markov chain defined by A might not be ergodic, but there is a unique stationary distribution which can be computed by PageRank.

PageRank with Restart: Algorithm[1]

Input: A directed graph G with n nodes (Web pages), $0 < \beta < 1, \epsilon > 0$.
Output: The PageRank vector r of the web pages in G.

Remove *dead ends* iteratively from G;
Build the stochastic matrix M_G (M for short);
Let $\pi^{(0)} = [\frac{1}{n}, \dots \frac{1}{n}]^T$
while (true) **do**
    $t = t + 1$;
    $\pi^{(t)} = A\pi^{(t-1)}$;
    If $||\pi^{(t)} - \pi^{(t-1)}||_1 < \epsilon$ **break**;
return $\pi^{(t)}$.

[1] see [1] for efficiency issues

Experimental Evaluation

Study [6] on community detection from seed sets.

Dataset	Nodes	Edges	Communities
DBLP	317080, authors	1049866, co-authorship	13477, conferences
Amazon	334863, products	925872, co-purchased	151037, product categories
YouTube	1134890, users	2987624, friendship	8385, user-defined groups

Figure: Datasets with ground-truth communities.

Experimental Evaluation: Settings

Consider the 600 communities[2] closest in size to $c_{\max}^{3/4}$.

Fair evaluation as communities have approximately the same size.

Recall$= \frac{|P \cap C|}{|C \setminus S|}$, where:

- P is the set of nodes found by the algorithm with $|P| = k$;
- C is the ground-truth community we wish to find;
- S is the set of seed nodes.

S is chosen to be a randomm subset of C with cardinality $\frac{|C|}{10}$.

[2] $c_{\max}$ = size of the largest community.

Experimental Evaluation: Results

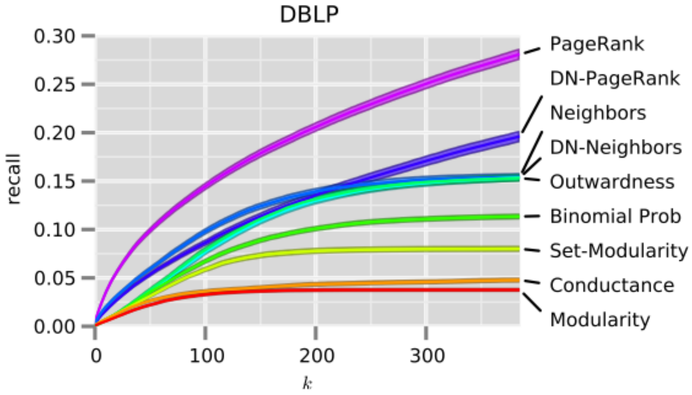

Figure: Recall as a function of k. Probability of jump in PR with restart $= 0.1$ ($\beta = 0.9$). The envelopes represent two standard errors centered about the mean.

Experimental Evaluation: Results

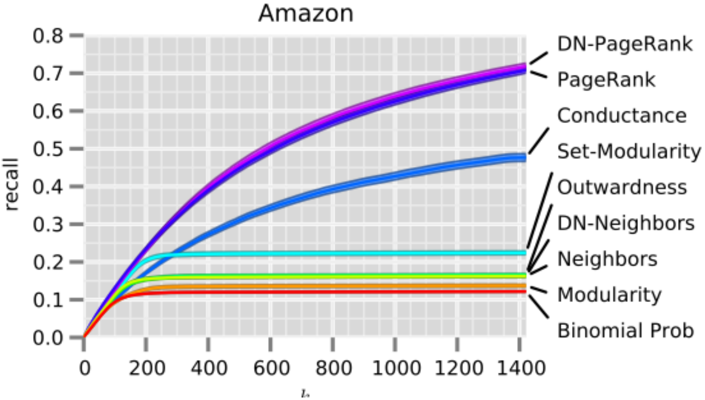

Figure: Recall as a function of k. Probability of jump in PR with restart $= 0.1$ ($\beta = 0.9$). The envelopes represent two standard errors centered about the mean.

Experimental Evaluation: Findings

Findings:

- PageRank with restart is simple and efficient and performs best.
- The PR algorithm needs to be iterated for 2-3 steps.

Limitations:

- how large k must be?
- good also with other choices of β, set of communities, datasets?

A Combinatorial Approach: Problem Definition ([7])

Problem Definition: Given a graph $G = (V_G, E_G)$, $S \subseteq V$, $d \in \mathbb{N}$ find an induced subgraph $H = (V_H, E_H)$ of G such that:

1. H is connected;
2. $S \subseteq V_H$
3. the distance between any node in S and $V_H \setminus S$ is at most d;
4. the minimum degree of H is maximized (among all subgraphs satisfying constraints 1-3).

A Combinatorial Approach: Algorithm ([7])

At each step $t = 1, \ldots, n$:

1. let $G_t = (V_t, E_t)$ be the current graph ($G_1 = G$).
2. If there is a node violating the distance constraint, remove it.
3. Otherwise, remove a node (and all its edges) with min. degree in G_t.

If none of the G_t's satisfy all the constraints return *unfeasible*.
Otherwise, among the subgraphs G_t's satisfying all the constraints, return the one with maximum minimum degree.

A Combinatorial Approach: Proof

Theorem 1

If there is a feasible solution, the previous algorithm computes an optimum solution otherwise it returns unfeasible.

Proof.

Let $O = (V_O, E_O)$ be an optimum solution (if any) and let $H = (V_H, E_H)$ be the graph returned by the algorithm. Let t be the first step when a node $v \in O$ is deleted from the current graph ($v \in V_t$). There must be such a step as we remove eventually all nodes. O is a subgraph of G_t, which implies that v satisfies the distance constraint in G_t. Therefore all nodes in G_t satisfy the distance constraint. It follows that:

$$\delta_{\min}(H) = \delta_{\min}(G_t) = \delta_{\min}(O).$$

□

References I

[1] Glen Jeh and Jennifer Widom.
Scaling personalized web search.
In *WWW*, pages 271279. ACM, 2003.

[2] Ingmar Weber, Venkata R Kiran Garimella, and Alaa Batayneh.
Secular vs. islamist polarization in egypt on twitter
ASONAM pages 290297. ACM, 2013.

[3] Aaron Clauset.
Finding local community structure in networks.
Physical review E, 72(2):026132, 2005.

[4] Alan Mislove, Bimal Viswanath, Krishna P Gummadi, and Peter Druschel.
You are who you know: inferring user profiles in online social networks.
In *WSDM*, pages 251260. ACM, 2010.

References II

[5] Alan Mislove, Bimal Viswanath, Krishna P Gummadi, and Peter Druschel.
You are who you know: inferring user profiles in online social networks.
In *WSDM*, pages 251260. ACM, 2010.

[6] Kloumann, Isabel M., and Jon M. Kleinberg.
Community membership identification from small seed sets.
SIGKDD, 2014.

[7] Sozio, M., and A. Gionis.
The community-search problem and how to plan a successful cocktail party.
ACM SIGKDD, 2010.