

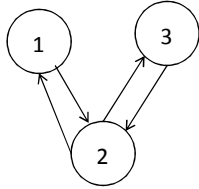
Network Structure of the Web

Part 2

Web Surfing as a Markov Chain Process, continued

- Adjacency Matrix \mathbf{A} :
 - If there is a hyperlink from page i to page j , then $A_{ij} = 1$, otherwise $A_{ij} = 0$.
- Transition Matrix \mathbf{P} :
 - If a row of \mathbf{A} has no 1s (i.e., no out-links), then insert $1/N$ for each element in that row in \mathbf{P} (uniform teleporting probability)
 - Otherwise, divide each 1 in the row in \mathbf{A} by the number of 1s in its row. (uniform probability of going to out-link)
 - Multiply the resulting matrix by $(1 - \alpha)$ (probability of going to that linked page by *not* teleporting)
 - Add α/N to every entry of the resulting matrix (probability of going to that each by teleporting)

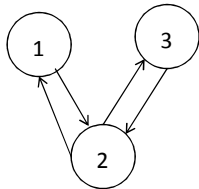
- Exercise 21.6: Consider the following web graph. What are the transition matrices for $\alpha = 0$ and 0.5?



Computing PageRank, continued

- Suppose $\alpha = 0.5$. Let \mathbf{x}_t be the probability distribution over the states at time t .

Suppose surfer starts in state 1. I.e., $\mathbf{x}_0 = (1 \ 0 \ 0)$.



$$\mathbf{P} = \begin{pmatrix} 1/6 & 2/3 & 1/6 \\ 5/12 & 1/6 & 5/12 \\ 1/6 & 2/3 & 1/6 \end{pmatrix}$$

After one time step, we have $\mathbf{x}_1 = \mathbf{x}_0 \mathbf{P} = (1/6 \ 2/3 \ 1/6)$

After two time steps, $\mathbf{x}_2 = \mathbf{x}_1 \mathbf{P} = (1/3 \ 1/3 \ 1/3)$.

Keep going. Finally reach steady state of $(5/18 \ 4/9 \ 5/18)$. **[Show this is a steady state]**

Questions

- What is the minimum possible PageRank of a page?
- How does varying α affect PageRank?

From <http://www.geek.com/articles/chips/googles-pagerank-algorithm-traced-back-to-the-1940s-20100217/>

Earlier forerunner to PageRank in the work of the Harvard economist Wassily Leontief:

“In 1941, Leontief published a paper in which he divides a country's economy into sectors that both supply and receive resources from each other, although not in equal measure. One important question is: what is the value of each sector when they are so tightly integrated? Leontief's answer was to develop an iterative method of valuing each sector based on the importance of the sectors that supply it. Sound familiar? In 1973, Leontief was awarded the Nobel Prize in economics for this work.”

Other Uses for Page Rank

- Ranking journal impact (nodes are journals, links are citations in articles in one journal to articles in the other journal -- e.g., see <http://www.eigenfactor.org>)
- Ranking doctoral programs (departments are nodes, one node links to another if it hires faculty from that dept.)
- Food webs – species that are essential to an ecosystem

Googling Food Webs: Can an Eigenvector Measure Species' Importance for Coextinctions?

Stefano Allesina^{1*}, Mercedes Pascual^{2,3,4}

¹ National Center for Ecological Analysis and Synthesis, Santa Barbara, California, United States of America, ² Department of Ecology and Evolutionary Biology, University of Michigan, Ann Arbor, Michigan, United States of America, ³ Santa Fe Institute, Santa Fe, New Mexico, United States of America, ⁴ Howard Hughes Medical Institute

Abstract

A major challenge in ecology is forecasting the effects of species' extinctions, a pressing problem given current human impacts on the planet. Consequences of species losses such as secondary extinctions are difficult to forecast because species are not isolated, but interact instead in a complex network of ecological relationships. Because of their mutual dependence, the loss of a single species can cascade in multiple coextinctions. Here we show that an algorithm adapted from the one Google uses to rank web-pages can order species according to their importance for coextinctions, providing the sequence of losses that results in the fastest collapse of the network. Moreover, we use the algorithm to bridge the gap between qualitative (who eats whom) and quantitative (at what rate) descriptions of food webs. We show that our simple algorithm finds the best possible solution for the problem of assigning importance from the perspective of secondary extinctions in all analyzed networks. Our approach relies on network structure, but applies regardless of the specific dynamical model of species' interactions, because it identifies the subset of coextinctions common to all possible models, those that will happen with certainty given the complete loss of prey of a given predator. Results show that previous measures of importance based on the concept of "hubs" or number of connections, as well as centrality measures, do not identify the most effective extinction sequence. The proposed algorithm provides a basis for further developments in the analysis of extinction risk in ecosystems.

Googling Food Webs: Can an Eigenvector Measure Species' Importance for Coextinctions?

Stefano Allesina^{1*}, Mercedes Pascual^{2,3,4}

¹ National Center for Ecological Analysis and Synthesis, Santa Barbara, California, United States of America, ² Department of Ecology and Evolutionary Biology, University of Michigan, Ann Arbor, Michigan, United States of America, ³ Santa Fe Institute, Santa Fe, New Mexico, United States of America, ⁴ Howard Hughes Medical Institute

Abstract

A major challenge in ecology is forecasting the effects of species' extinctions, a pressing problem given current human impacts on the planet. Consequences of species losses such as secondary extinctions are difficult to forecast because species are not isolated, but interact instead in a complex network of ecological relationships. Because of their mutual dependence, the loss of a single species can cascade in multiple coextinctions. Here we show that an algorithm adapted from the search engines, providing the gap between the simple and secondary specific models, do not exist in the network.

“Here we show that an algorithm adapted from the one Google uses to rank web-pages can order species according to their importance for coextinctions, providing the sequence of losses that results in the fastest collapse of the network.”

Hubs and Authorities (HITS Algorithm)

- Proposed by Jon Kleinberg (Cornell) at same time Brin and Page were developing PageRank
- HITS: “Hyperlinked-induced topic search”
- Supposedly used by Teoma and Ask.com

Hubs and Authorities Main ideas

Each node has a **hub** score and an **authority** score

Hub: Web site that points to a lot of good authorities

Authority: Web site that is pointed to by a lot of good hubs

Circular definition \longrightarrow iterative computation

$$h(v) \leftarrow \sum_{y: v \mapsto y} a(y)$$

$$a(v) \leftarrow \sum_{y: y \mapsto v} h(y)$$

where $a \mapsto b$ means a links to b

Hubs and Authorities How to compute scores

Let \mathbf{h} = vector of hub scores of web pages in collection

Let \mathbf{a} = vector of authority scores of web pages in collection

Let \mathbf{A} denote the adjacency matrix of the web pages in collection

Then:

$$\mathbf{h} \leftarrow \mathbf{A}\mathbf{a} \quad \text{and} \quad \mathbf{h} \leftarrow \mathbf{A}\mathbf{A}^T\mathbf{h}$$

$$\mathbf{a} \leftarrow \mathbf{A}^T\mathbf{h} \quad \mathbf{a} \leftarrow \mathbf{A}^T\mathbf{A}\mathbf{a}$$

Can turn these into eigenvector equations:

$$\lambda_i \mathbf{h} = (\mathbf{A}\mathbf{A}^T)\mathbf{h}$$

$$\lambda_i \mathbf{a} = (\mathbf{A}^T\mathbf{A})\mathbf{a}$$

Hubs and Authorities

How to compute scores, continued

- HITS Algorithm:
 - Assemble the target subset of web pages from a given query, form the graph induced by their hyperlinks, and compute \mathbf{AA}^T and $\mathbf{A}^T\mathbf{A}$
 - Compute the principal eigenvectors of \mathbf{AA}^T and $\mathbf{A}^T\mathbf{A}$ to form the vector of hub scores \mathbf{h} and the vector of authority scores \mathbf{a} .
 - Output the top-scoring hubs and top-scoring authorities

Choosing the subset of the web for the HITS algorithm (One proposed method, described in textbook)

1. Given a query , use a text index to get all pages containing the terms of the query. Call this the *root* set of pages.
2. Build the *base* set of pages to include the root set as well as any page that either links to a page in the root set, or is linked to by a page in the root set.

Use the base set to compute hub and authority scores.

Differences between PageRank and HITS

Illustration of Web Structures for PageRank and HITS

How can PageRank and HITS be spoofed /
manipulated?

Web Spam Detection Using Network Structure

(Castillo et al., 2006, Yahoo Research Barcelona)

- Web spam = manipulation of content or link structure to get pages rank higher on search engines
- PageRank (used alone) is highly susceptible to Web spam.
- **Example from 2004 paper:** 100 million pages were ranked using PageRank. 11 of top 20 were pornographic pages, which achieved high ranking through link manipulation.

Types of Web Spam

- **Link spam:** Creation of tightly knit collection of links aimed at affecting the outcome of a link-based ranking algorithm.
- **Content spam:** “Malicious” crafting the content of web pages
- **Cloaking:** Sending different content to a search engine than to regular web site visitors.

Approach of Castillo et al. paper

- Used labeled data set with 78 million pages and 3 billion edges.
- Used machine learning classification approach with link-based features to describe nodes

Link-based features

Link-based features

- In-degree, out-degree
- Edge reciprocity
- Assortativity (ratio between degree of node and average degree of neighbors)
- PageRank of node and relationships to PageRank of in-link neighbors
- Truncated PageRank (how much PageRank it gets from close neighbors)

- TrustRank: If page has high PageRank but no relationship with set of known trusted pages, likely to be spam.
- SpamMass: amount of PageRank received from a spammer.
- Estimation of d-supporters (set of all pages that have shortest path of length d from page) -- to see extent to which pages are isolated from rest of Web graph.

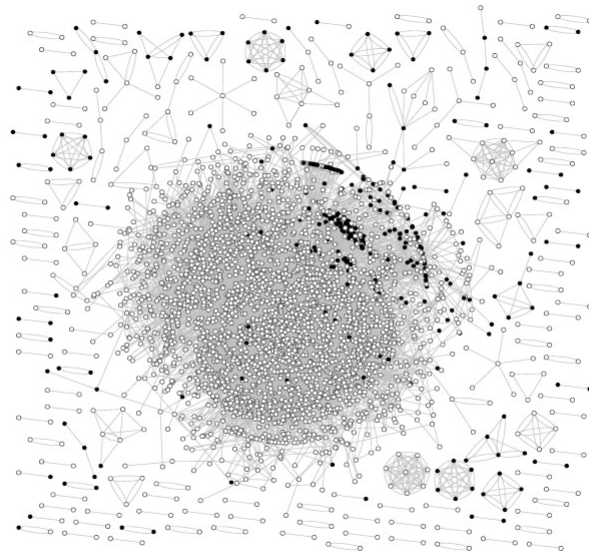
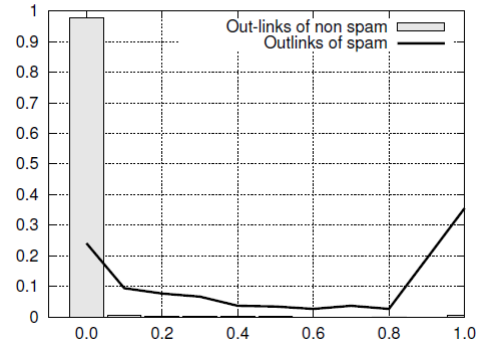
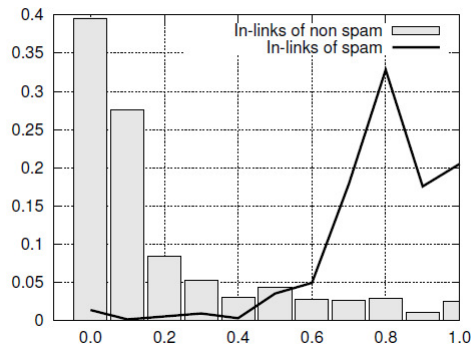


Figure 7: Graphical depiction of the hostgraph (undirected), pruned to include only labeled nodes with a connection of over 100 links between them. Black nodes are spam, white nodes are non-spam. Most of the spammers in the larger connected component are clustered together (upper-right end of the center portion). Most of the other connected components are single-class (either only spam nodes, or only non-spam nodes).

(a) Fraction of spam nodes in out-links



(b) Fraction of spam nodes in in-links



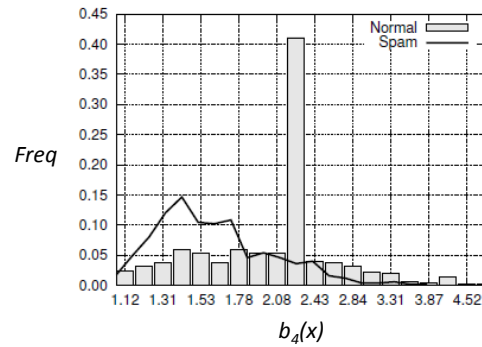


Figure 3: Histogram of the minimum ratio change of the # of neighbors from distance i to distance $i-1$

$$b_d(x) = \min_{j \leq d} \left\{ \frac{|N_j(x)|}{|N_{j-1}(x)|} \right\}$$

where $N_d(x)$ is the set of d -supporters of page x
 y is a d -supporter of x if shortest path from y to x is d .

- Several content-based features as well.

Table 4: Comparing link and content features

	Both	Link-only	Content-only
True positive rate	78.7%	79.4%	64.9%
False positive rate	5.7%	9.0%	3.7%
F-Measure	0.723	0.659	0.683

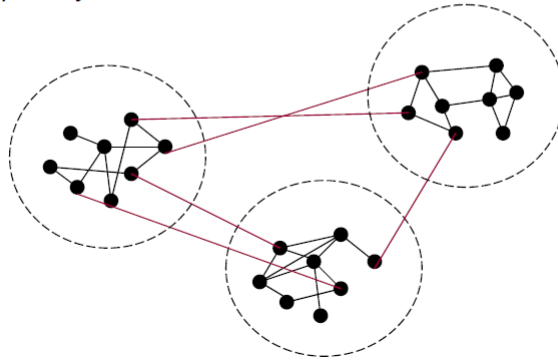
Community and social network detection

(slides adapted from www.phys.psu.edu/~ralbert/phys597_09-fall/c04_netw_commun.pdf)

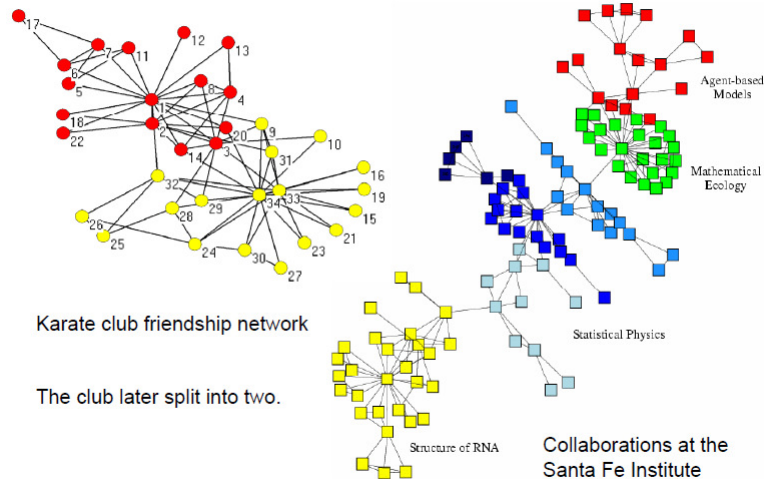
From www.phys.psu.edu/~ralbert/phys597_09-fall/c04_netw_commun.pdf

Community structure in networks

- Many real-world networks, especially social ones, exhibit **community structure** (also called **modularity**).
- Intuitively community structure can be defined as the existence of subgraphs that are densely connected but sparsely inter-connected.



From www.phys.psu.edu/~ralbert/phys597_09-fall/c04_netw_commun.pdf



From www.phys.psu.edu/~ralbert/phys597_09-fall/c04_netw_commun.pdf

Examples of communities

- Network : World Wide Web
 - Nodes : webpages
 - Edges : hyper-references
- Communities : Nodes on related topics
- Network : Friendship network
 - Nodes : people
 - Edges : friendship
- Communities : Group formation among people
- Network : Metabolic networks
 - Nodes : metabolites
 - Edges : participation in a chemical reaction
- Communities : Functional modules

From www.phys.psu.edu/~ralbert/phys597_09-fall/c04_netw_commun.pdf

Definitions of a community

- Cliques (completely connected subgraphs)
- Chain of cliques – adjacent cliques share every node except one
- k-clan – diameter (largest path length) is $\leq k$
- Definitions using the edges inside and outside a presumed community
 - k_i^{in} – edges of node i that stay inside the community
 - k_i^{out} – edges of node i that go outside of the community
 - Strong community: $k_i^{in} \geq k_i^{out}$ for every node i in the community
 - Weak community: $\sum_i k_i^{in} \geq \sum_i k_i^{out}$, where the sum is over nodes in the community

F. Radicchi et al., PNAS 101, 2658 (2004).

From www.phys.psu.edu/~ralbert/phys597_09-fall/c04_netw_commun.pdf

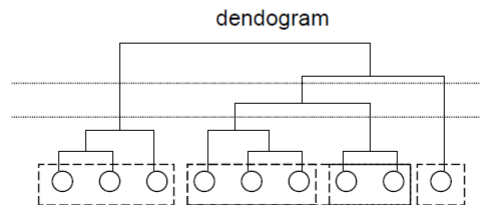
Community Detecting Algorithms

- **Input:** A network $G(n,m)$
- **Output:**
 - The number of communities
 - Classification of nodes into these communities
- Two families of methods:
 - Agglomerative (bottom up)
 - Divisive (top down)

From www.phys.psu.edu/~ralbert/phys597_09-fall/c04_netw_commun.pdf

Agglomerative method: hierarchical clustering

- Calculate a weight (connectivity measure) W_{ij} for every pair i, j of vertices
 - Example of weight: number of node-independent paths between i and j .
- Start with each node as a separate community
- Unite the highest-weight node pair(s)
- Calculate the weights between the newly formed community(ies) as averages over the nodes in the community
- Repeat



From www.phys.psu.edu/~ralbert/phys597_09-fall/c04_netw_commun.pdf

Divisive method: betweenness centrality algorithm

- **Betweenness centrality** of an edge is the number of shortest paths between pairs of vertices that run along it
 - **Algorithm:**
 - Calculate the betweenness for all edges in the network
 - Remove the edge with highest betweenness
 - Recalculate the betweenness for all edges affected by the removal
 - Repeat
- M.E.J. Newman, Phys. Rev. E 69, 066133, 2004.

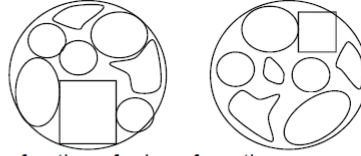


This algorithm also leads to a dendrogram

From www.phys.psu.edu/~ralbert/phys597_09-fall/c04_netw_commun.pdf

Strength of communities

To check if a particular division is meaningful, we can determine

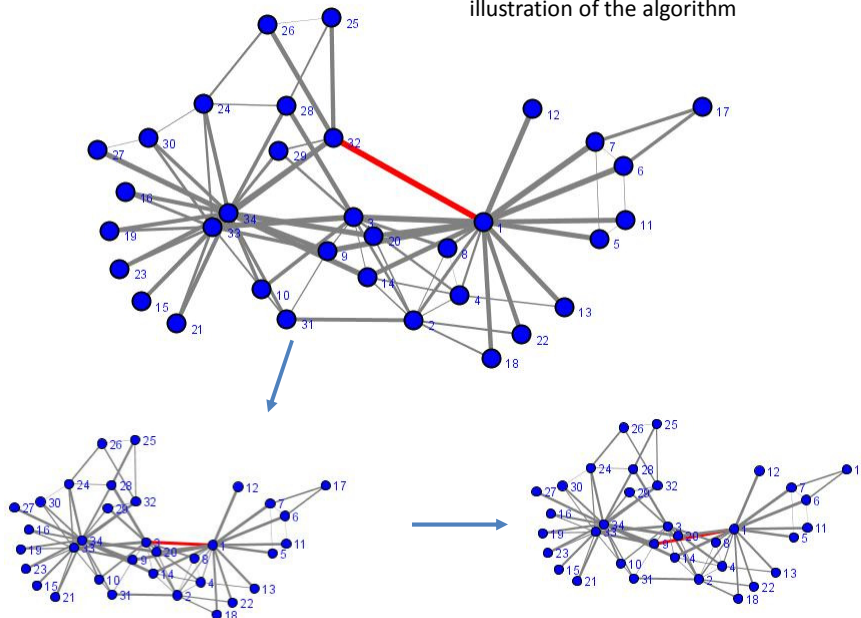


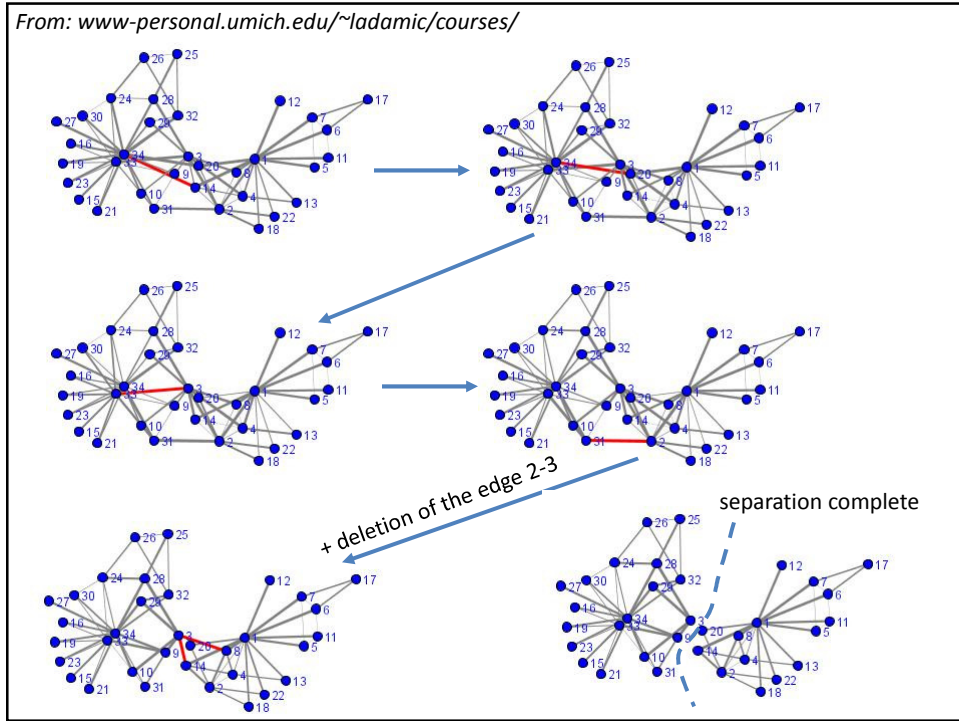
1. The fraction of edges within the community divided by the fraction of edges from the community to outside of the community
2. The observed number of edges within the community divided by the expected number of edges within a community if the edges are distributed randomly
3. Modularity measure Q , defined as the fraction of edges that fall within communities, minus the expected value of the same quantity if edges fall at random without regard for the community structure.

For either measure, the higher the result, the better the proposed community structure. M. Girvan and M.E.J. Newman, PNAS 99 (2002).

From: www-personal.umich.edu/~ladamic/courses/

illustration of the algorithm





From: www-personal.umich.edu/~ladamic/courses/

Email as Spectroscopy: Automated Discovery of Community Structure within Organizations

- Joshua R. Tyler, Dennis M. Wilkinson, Bernardo A. Huberman
Communities and technologies (2003)
- Modifications of Girvan-Newman betweenness clustering algorithm
 - stopping criterion: stop removing edges before disconnecting a leaf node

smallest graph w/ 2 viable communities

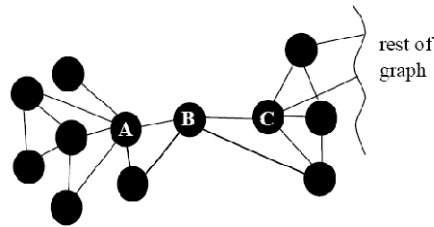
cut is not made

- randomness is introduced by calculating shortest paths from only a subset of nodes and running the entire algorithm several times
 - nodes that border several communities fall in different communities on different runs
 - distinguishes between brokers and single-community nodes

From: www-personal.umich.edu/~ladamic/courses/

inter-community nodes

- Example of network structure, where one node B, could arguably belong to either community
- With “noisy” algorithm, can keep track of % of time B ends up in A’s community or C’s community



From: www-personal.umich.edu/~ladamic/courses/

email spectroscopy: results

- data: HP labs email network (~ 400 nodes, 3 months, mass mailings removed, 30 message threshold)
- giant component of 434 nodes
- 66 communities, 49 correspond exactly to organizational units
- other 17 contain individuals from 2 or more organizational units within the company
- Field interviews confirmed accuracy of algorithm: individuals identified their communities, divisions in formal groups, and overlaps in interest on joint projects

Example community	
Individual	Strength in community (max 50)
Person 34	50
Person 267	50
Person 56	50
Person 406	50
Person 212	50
Person 246	29
Person 331	15
Person 87	7