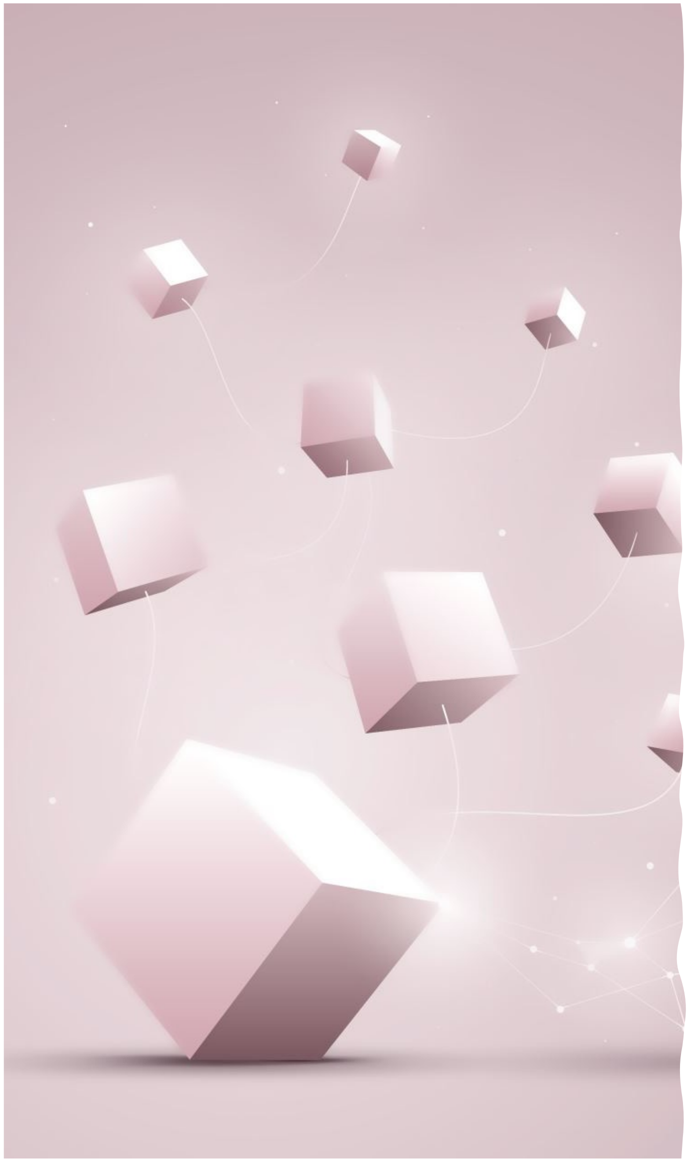


Email as Spectroscopy: Automated Discovery of Community Structure within Organizations

Presenter: Ye Yuan

2023.03.27

Outlines

- Paper and Authors Background
- Problem Formulation
- Methodology
- Experiments
- Results and Conclusion

Paper and Authors Background

- Published on **The Information Society, Volume 21, 2005 Issue 2**
- Received 16 Mar 2004
- Accepted 30 Jun 2004
- Published online: 24 Feb 2007
- This work was done by HP Labs
- Joshua R. Tyler: Researcher at HP Labs
- Dennis M. Wilkinson: Researcher at HP Labs
- Bernardo A. Huberman: Director of HP Labs

Motivation

- **Communities of practice:** the informal networks of collaboration that naturally grow and coalesce within the organization.
- These communities can be used to uncover the reality of how people find information and execute their tasks.
- These informal networks coexist with the formal structure of the organization and serve many purposes, such as resolving the conflicting goals of the institution to which they belong, solving problems in more efficient ways, furthering the interests of their members, and enhancing the productivity of the formal organization.

Problem

- Due to the mentioned values of communities of practice, a fast and accurate method of identifying them is desirable.
- Previous works mainly gathered data from interviews, surveys, or other fieldwork and constructed links and communities by manual inspection. Accurate but time-consuming and labor-intensive.

This paper’s solution

- Uses Email data to construct a network of correspondences and then discovers the communities by partitioning this network in a particular way.
- Only use the names of the senders and receivers, which minimizes privacy issues.

Methodology

- The methods consist of two basic steps:
  - The first one uses the headers of email logs to construct a graph where the vertices are senders or recipients of email messages, and the links denote a direct email between the nodes they connect.
  - The second step uses the algorithm we will introduce later to find the communities embedded in the graph.

Graph Construction through the Email Logs

- Vertices: People, i.e., the senders and receivers shown in the email logs.
- Edges: connect two vertices if the number of emails passed between them is larger than or equal to the threshold.
- Threshold: in this paper, the authors used a threshold of 30 messages.
- Undirected Graph: 单向至少5messages，潜在的条件

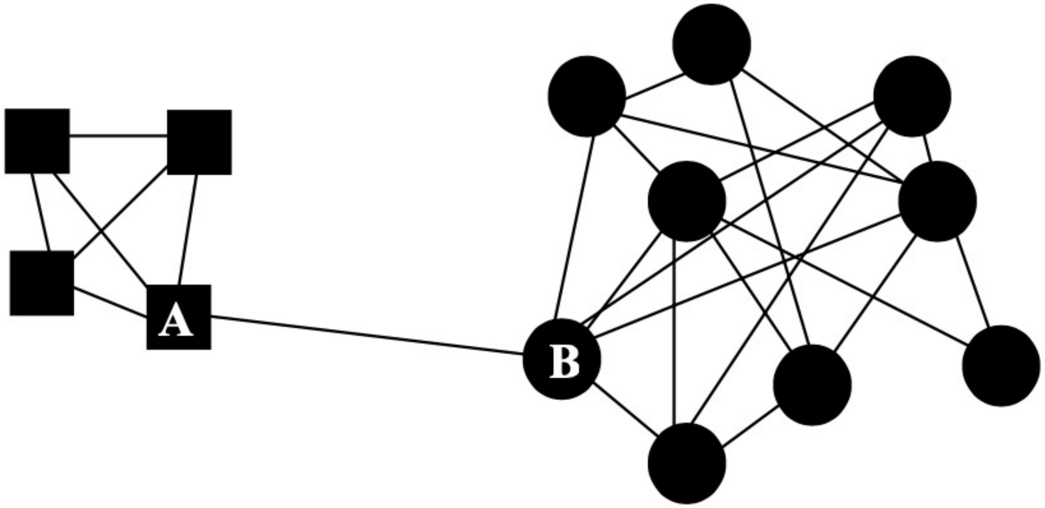
The proposed algorithm

- Idea:
  - Remove a “certain” edge from the graph so that a giant connected component will be partitioned into two separate connected components.
  - Keep removing edges until we find some communities of practice.

- Problems:
  - How should we define the community structure in a graph?
  - How should we choose such a “certain” edge?
  - How should we define the stop criteria (i.e., when should we stop partitioning further)?

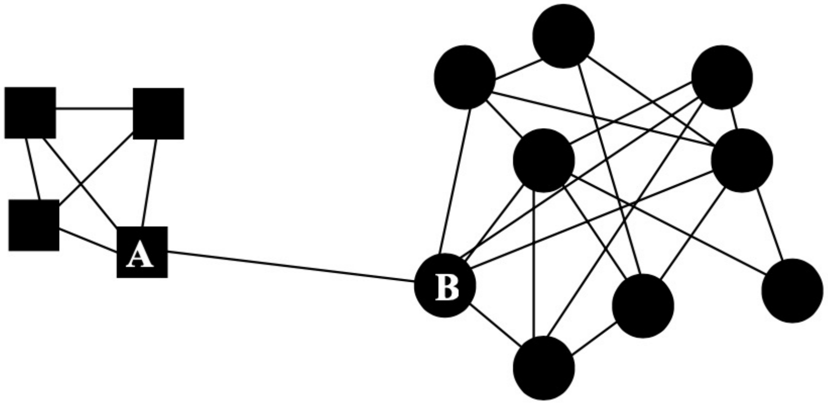
How should we define the community structure in a graph?

- A graph has a community structure if it consists of subsets of vertices, with many edges connecting vertices of the same subset but few edges lying between subsets.



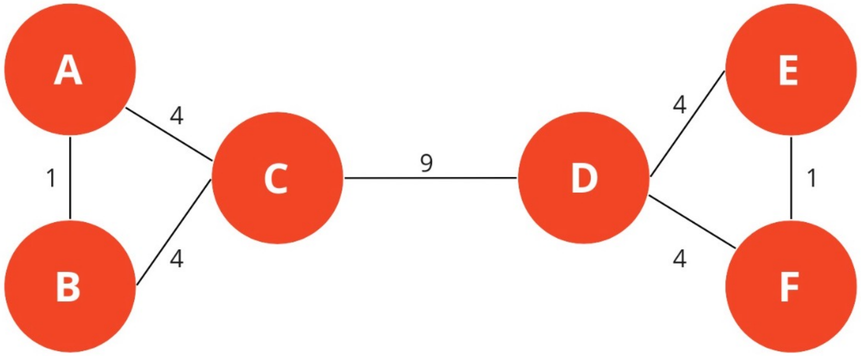
How should we choose such a “certain” edge?

- The edge connecting A and B is called an inter-community edge. The other edges are called intra-community edges correspondingly.
- Our goal is to remove the inter-community edge so that the whole connected graph will be partitioned into two separate communities correctly.



How should we choose such a “certain” edge?

- We need a quantitative way to identify the inter-community edges.
- This paper exploited Freeman’s [2] notation of betweenness to find inter-community edges.
- The betweenness of an edge is defined as the number of shortest paths that traverse it.
- The inter-community edges link many vertices in different communities and have a high betweenness. In contrast, the betweenness of the intra-community edges is low.

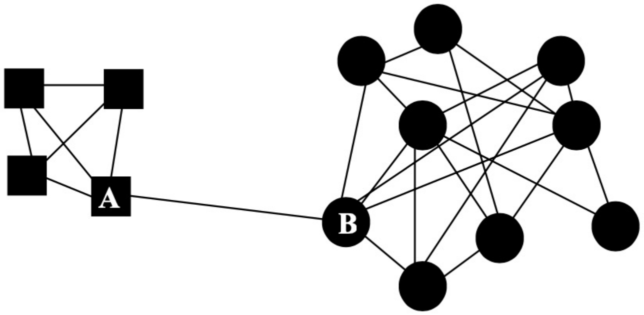


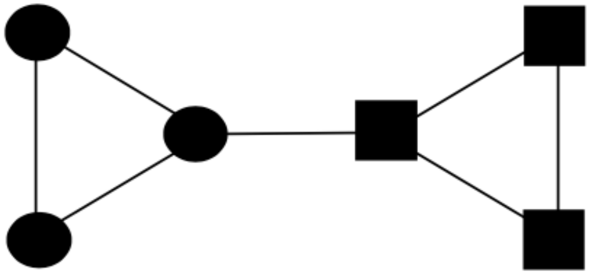
Betweenness Calculation

- To fast calculate the betweenness of all edges in the graph, this paper adopts the algorithm of Brandes[3].
- Consider the shortest paths between a single vertex, which is called the “center”, and all other vertices.
- Calculate the betweenness of each edge based on these shortest paths and add them to a running total.
- Then change centers and repeat until every vertex has been the center once.
- The running total for each edge is then equal to exactly twice the exact betweenness of that edge.

How should we define the stop criteria (i.e., when should we stop partitioning further)?

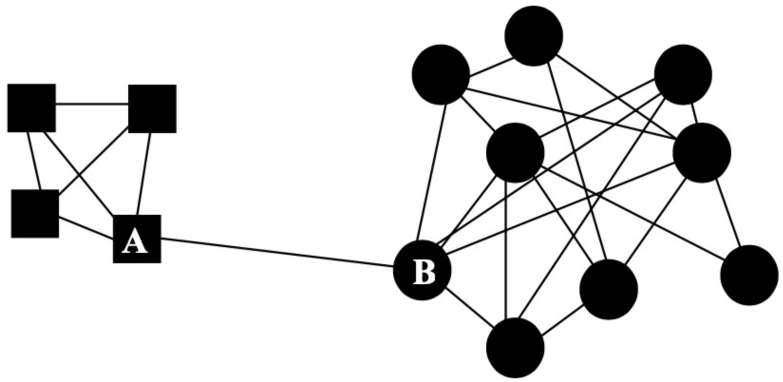
- We should stop removing edges when we cannot further meaningfully subdivide the communities.
- Structurally, a component of 5 or fewer vertices cannot consist of two viable communities.
- **If at any time we remove an edge from our graph and separate a component of size smaller than 6, we can identify it as a community.**

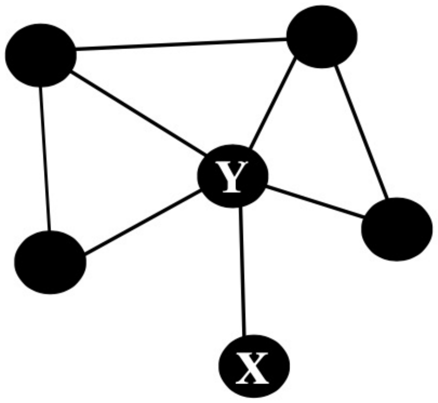




How should we define the stop criteria (i.e., when should we stop partitioning further)?

- Components of size equal to or greater than six can also be individual communities.
- In general, the single edge connecting a leaf vertex to the rest of a graph of N vertices has a betweenness of N – 1, since it contains the shortest path from X to all N – 1 other vertices.
- **The stopping criterion for components of size equal to or greater than six is therefore that the highest betweenness of any edge in the component be equal to or less than N – 1.**





Multiple Community Structures

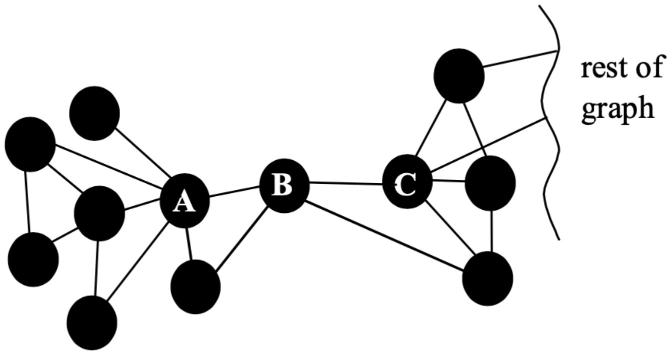
- The removal of any one edge affects the betweenness of all the other edges. Therefore, the order of removal of edges affects which edges are removed.
- Early in the process, there are many inter-community edges which have high betweenness and the choice of which to remove is arbitrary but dictates which edges will be removed later.
- We can take advantage of this arbitrariness to repeatedly partition the graph into many different sets of communities and then compare the different sets and aggregate the result into a final list of communities.

Multiple Community Structures

- Consider the placement of John and Sarah in communities.
- If John appears within the same community in all 50 sets, it is clear that John definitely belongs to that community. The order of edge removal had no effect on him.
- However, if Sarah appears in one community in some sets in another (or even several others) in other sets, the order of edge removal did affect her. We should consider that she has some affiliation with those two (or more) communities.
- If we only considered one community structure, Sarah would have been placed in one community rather than several communities, and we would have lost information about her role in the other community (communities).

Multiple Community Structures

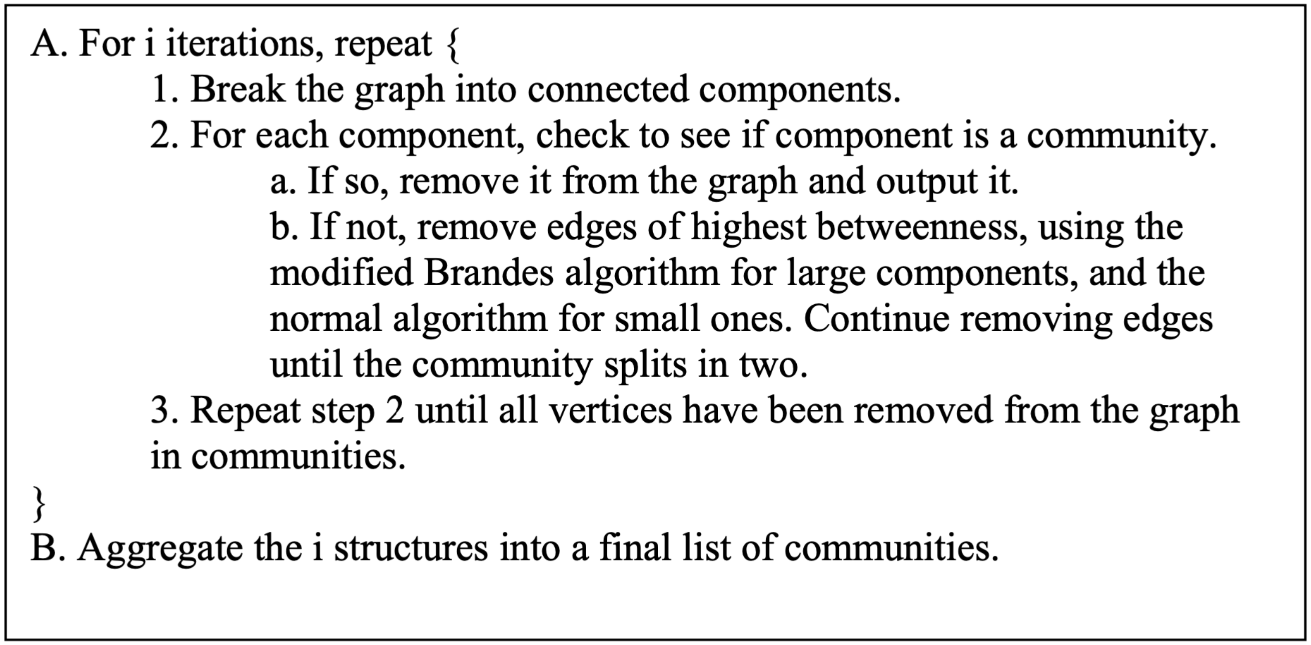
- The graph consists of two communities, one on the left, including vertex A, and another on the right, including C.
- BC initially has the highest betweenness among its edges, and AB's betweenness is also high. If we choose to remove BC first, AB becomes an intra-community edge with low betweenness, which will never be removed, and vertex B will eventually be placed in a community with vertex A.
- If we removed AB first, BC would have been rendered intra-community, and vertex B would end up in the community with C.



Multiple Community Structures

- From all edges with high betweenness, we randomly select one to start the proposed algorithm.
- After applying the proposed algorithm n times, we obtain n community structures imposed on the graph. We can then compare the different structures and identify communities.
- For example, after imposing 50 structures on our graph, we might find: a community of people A, B, C, and D in 25 of the 50 structures; a community of people A, B, C, D, and E in another 20; and one of the people A, B, C, D, E and F in the remaining 5.
- This result can be shown in the following way:
- A(50) B(50) C(50) D(50) E(25) F(5)
- which signifies that A, B, C, and D form a well-defined community, E is related to this community but also to some other(s), and F is only slightly, possibly erroneously, related to it.

Pseudo-Code of the proposed algorithm



```
A. For i iterations, repeat {
        1. Break the graph into connected components.
        2. For each component, check to see if component is a community.
                a. If so, remove it from the graph and output it.
                b. If not, remove edges of highest betweenness, using the
                modified Brandes algorithm for large components, and the
                normal algorithm for small ones. Continue removing edges
                until the community splits in two.
        3. Repeat step 2 until all vertices have been removed from the graph
        in communities.
}
B. Aggregate the i structures into a final list of communities.
```

Experiments

- An experiment of the proposed algorithm was conducted by using email data from the HP Labs mail server.
- Starting from an original set of 878,765 logged emails over the period 25 November 2002 to 18 February 2003, the authors constructed a “clean” subset of 185,773 emails between any two of the 485 current HP Labs employees.
- Emails that had an external origin or destination are neglected for privacy issues.
- Messages sent to a list of more than 10 recipients are also ignored, as these emails were often lab-wide announcements.

Results and Conclusions

- Sixteen individuals are interviewed in seven different communities.
- All sixteen subjects gave positive affirmation that the community reflected reality. More specifically, eleven described the group as reflecting their department, four described it as a specific project group, and one said it was a discussion group on a particular topic.
- Nine of the sixteen (56.25%) said nobody was missing from the group, six people (37.5%) said one person was missing, and one person (6.25%) said two people were missing. Conversely, ten of the sixteen (62.5%) said that everybody in the group deserved to be there, whereas the remaining six (37.5%) said that one person in the group was misclassified.

References

- [1] J. R. Tyler, D. M. Wilkinson, and B. A. Huberman, Email as Spectroscopy: Automated Discovery of Community Structure within Organizations. arXiv, 2003. doi: 10.48550/ARXIV.COND-MAT/0303264.
- [2] Freeman, L. C. (1977). A set of measures of centrality based on betweenness. Sociometry, 40(1), 35–41. https://doi.org/10.2307/3033543
- [3] Ulrik Brandes (2001) A faster algorithm for betweenness centrality, The Journal of Mathematical Sociology, 25:2, 163-177, DOI: 10.1080/0022250X.2001.9990249

Thank you

Comments and Q&A