A Strategy to Build Connection with a Stranger Based on Social Network Topology

Shaojun Zhang, *Department of Statistics, University of Florida*

***Abstract*—As we know, people are connected in various social networks such as phone calls, emails and Facebook. Then an interesting question arises whether we can improve our chance of making friends with some particular stranger in the network? Here the stranger can be our potential spouse, colleague, advisor or boss. Of course we can directly make a phone call, send an email or send an invitation through Facebook to the stranger. But we are very likely to be ignored or reported since we are completely unknown to the stranger. Moreover, private information such as email addresses and phone numbers are unavailable most of time.**

One straightforward solution is to link one by one along the shortest path to the target. However, the problem becomes complicated when the shortest path is still long or when there exists more than one shortest path, especially in a dense network. In this report, I develop a strategy based on network topology including shortest paths, community structure and eigenvector centrality to deal with these problems. The method and the accompanying analysis will be illustrated on a dataset covering email communication of about 150 users of Enron, an American energy company.

I. Introduction

NOWADAYS, people have many ways to communicate with each other. We can make phone calls, write text messages, send emails as well as send messages online via websites or applets. All of these communications forms our social network.

An interesting problem is that if we have access to the network data, how we can benefit ourselves by building connection with some stranger that we are interested. For example, the stranger can be our potential advisor that we find on the department website when applying for a PhD program. There are many more such examples and this situation is very common.

Obviously, if there is a path between the stranger and me in the network, there must be a direct friend of mine on the path. Therefore, I can ask him or her to introduce the next person on the path to me. In this way, I can make friends one by one along the path until I reach the stranger. If there are many paths to the stranger, we will of course choose the shortest one. But what if there are still many shortest paths? Also, we may not be willing to keep connecting if the path is long.

In this report, I proposed a strategy that combines community structure, eigenvector centrality and some other criteria to deal with these problems. I set an upper bound for the number of people allowed to link and use it to truncate all shortest paths. All comparisons among candidate paths are based on these truncated paths. Louvain method and label propagation algorithm are used for community detection. Two versions of community score and an importance score are defined and used for path selection.

The outline of the report is as follows. In Sec. II, I describe the basic characteristics of the Enron network, some assumptions needed and the detailed method. In Sec. III, I give and analyze the main results. In Sec. IV, I describe several algorithms related to my strategy. In Sec. V, I give my conclusions, remaining problems and possible extensions.

II. Description

A. Data

The Enron email dataset [1] was collected and prepared by the CALO project. It contains data from about 150 users, mostly senior management of Enron organized into folders. This data was originally made public, and posted to the web, by the Federal Energy Regulatory Commission during its investigation. This data is valuable because it is the only substantial collection of “real” email that is public [2].

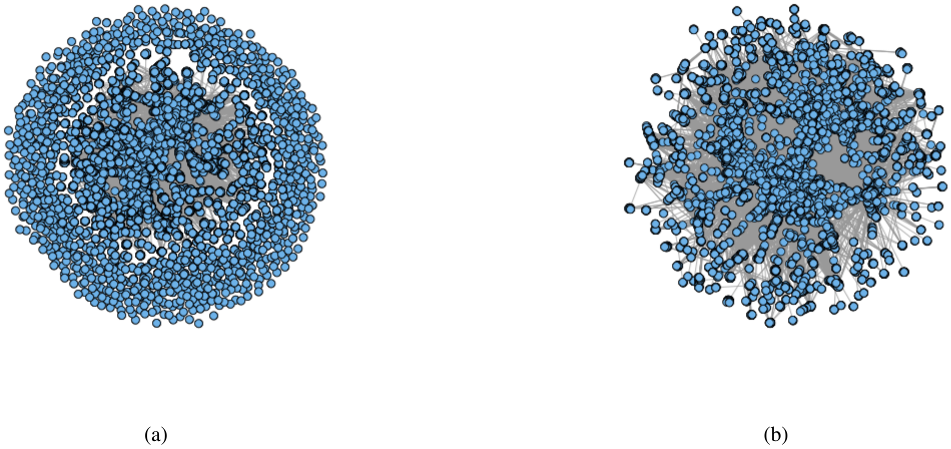
The network formed by these email communications is a typical social network. Its properties may be very different from those of other types of social network such as phone calls. But the method in this report can be applied to all networks including large datasets such as Facebook or Twitter.

The undirected graph induced by the entire Enron email dataset contains 36692 vertices and 183831 edges. Unfortunately, it is not a connected graph, which means that there does not exist a path between some pairs of users. Therefore, these pairs of users cannot become friends based on the information from the email communications.

However, we can focus on connected components so that any two users in each component are linked by at least one path. Among all connected components, there is a giant component which contains $N_v = 33696$ vertices and $N_e = 180811$ edges, covering 92% vertices and 98% edges respectively in the whole graph as shown in Fig. 1. Therefore, all the analysis and results below are based on the giant component instead of the whole graph.

The diameter of the graph is 13 and the average path length of the graph is 4.02. According to the well-known six degrees of separation theory, everyone and everything is six or fewer steps away, by way of introduction, from any other person in the world [3]. Therefore, the average distance between users in the Enron network is shorter than that in our real world. The Enron network may even be a small-world network [4].

The average degree of the graph is 10.73 and the average clustering coefficient of the graph is 0.71. To see the degree behavior for all vertices in the graph, a power law distribution with two parameters is fitted.



(a)

(b)

Fig. 1. The network of the Enron email communication. (a) is the whole graph and (b) is the subgraph of the giant component.

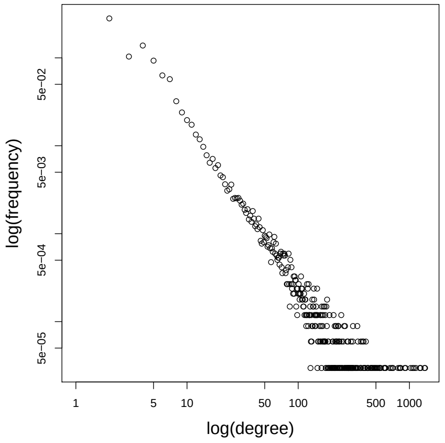


Fig. 2. Degree distribution.

First the probability density function of degree is assumed to be $f_d = Cd^{-\alpha}$ for d in $[x_{min}, \infty)$, where C is a constant. After integrating both sides with respect to d, we have $C = (\alpha - 1)x_{min}^{\alpha-1}$. Thus the density function becomes

$$f_d = \frac{\alpha - 1}{x_{min}} \left(\frac{d}{x_{min}} \right)^{-\alpha}.$$

After taking the log of both sides, we have a linear form

$$\log f_d = \log(\alpha - 1) + (\alpha - 1)\log x_{min} - \alpha \log d,$$

which seems suitable for the Enron dataset as shown in Fig. 2. The two parameters are estimated using maximum likelihood and we have $\hat{\alpha} = 1.98$ and $\widehat{x_{min}} = 5$.

Closeness centrality, betweenness centrality and eigenvector centrality for all vertices are calculated. Top 10 vertices with highest scores for each type of centrality are compared and shown in Table I. The results of closeness centrality and eigenvector centrality are similar while the results of betweenness centrality are much different.

TABLE I
TOP 10 VERTICES FOR THREE CENTRALITIES

Rank	Closeness	Betweenness	Eigenvector
1	137	5025	137
2	77	141	196
3	47	567	77
4	141	589	371
5	371	1140	1029
6	293	274	274
7	196	459	735
8	735	47	417
9	176	1029	176
10	417	293	293

B. Assumptions

To establish the method, I make the following assumptions:

1) Friendships are based on edges in the graph and each user can only make new friends by being introduced. Since the network dataset is all we have, we can only deduce friendships and possible communications among users from the network topology.
2) Users in the same community have a closer relationship than those from different communities. By the definition of community, a vertex in a community is densely connected to other vertices in the same community, and sparsely connected to vertices outside the community. Therefore, in our case two users in the same community are likely to share more common friends and have more common interests.
3) The more importance a user has, the more likely his or her friends will be introduced successfully. This assumption is motivated by the fact that if a user has a large influence, his or her friends will be more willing to accept or believe what the user says.
4) Each user can only make at most 3 new friends excluding his or her direct friends. If the stranger becomes a

friend of the user, the connection is built successfully. This restriction is based on the fact that each link has cost in the form of time, money or something else.

5) The network is fixed. That is to say, even if some user has made new friends, the network will remain unchanged. This assumption is made for simplicity. If several edges are added to a large network, its topology usually will not change much. What's more, while one user is making new friends, the relationships among other users are changing as well.

C. Method

To apply the method, we need to choose two users, one as the user who is going to connect and the other as the stranger. They can be any two users in the network. After that, there are three steps to find the "best" path between them.

The first step is to find all shortest paths. Common shortest path algorithm such as Dijkstra's algorithm can only find one shortest path. Therefore, Yen's algorithm is used here to find K-shortest paths. As long as K is large enough, we can find all shortest paths in the graph. For the Enron network, K is chosen to be 10.

If the length of shortest paths $d \leq 4$, the user can reach the stranger successfully according to Assumption 1 and Assumption 4. Otherwise, the stranger is not reachable. In this case, the problem of the user becomes how to get as close as possible to the stranger so that the user can obtain a largest amount of possible information and may find other ways to reach the stranger. Since the user can reach at most 4 users on the path, all shortest paths are truncated if $d > 4$.

Next, I consider the community structure in the graph. Louvain method and label propagation algorithm are used for community detection. Two versions of community score are defined as follows:

$$\begin{aligned} C_1 = &\text{ number of pairs of neighboring users on the path} \\ &\text{ that belong to the same community,} \\ C_2 = &\text{ number of users on the path that belong to the} \\ &\text{ community containing the stranger.} \end{aligned}$$

By Assumption 2, a high community score indicates a high probability of connecting the stranger successfully. Therefore, the paths with highest community score (C_1 or C_2) are selected as further candidates.

The importance of users in the network is used for final decision. An importance score is defined as follows:

$$I = \text{ total eigenvector centrality scores on the path.}$$

By Assumption 3, a high importance score also indicates a high probability of successful connections. Therefore, the path with highest importance score (I) is selected as the best path.

If there is only one path left after the first or the second step, the path is picked as the best path. If there are more than one path left after the last step, the best path is chosen randomly from them. The algorithm is as follows.

```
Select all shortest paths using Yen's algorithm.
**if** $d > 4$ **then**
    All shortest paths are truncated to first four vertices.
**end if**
**if** only one path left **then**
    Return the path.
**else**
    Detect communities using Louvain method or label
    propagation and select the paths with largest $C_i$, $i =$
    1 or 2.
    **if** only one path left **then**
        Return the path.
    **else**
        Compute eigenvector centrality scores and return the
        path with largest $I$.
    **end if**
**end if**
```

III. Evaluation

For community detection, 239 communities are found by Louvain method and 903 by label propagation algorithm. Fig. 3 shows the distribution of community sizes for these two methods. Most communities found by label propagation algorithm contain a very small number of users.

10 pairs of users are randomly sampled from the network with the restriction that the two users in any pair belong to different communities. The restriction is made to bring some challenge to connections and to resemble real-life situations when we are interested in linking someone who is not close to us.

The results are presented in Table II and Table III. In these two tables, the first two columns contain the indices of users who are going to connect and the indices of strangers. SP# means the number of shortest paths between two users. Column C_1 and C_2 contain the best paths found with community score defined as C_1 and C_2 respectively. Success indicates whether the user has build connection with the stranger successfully.

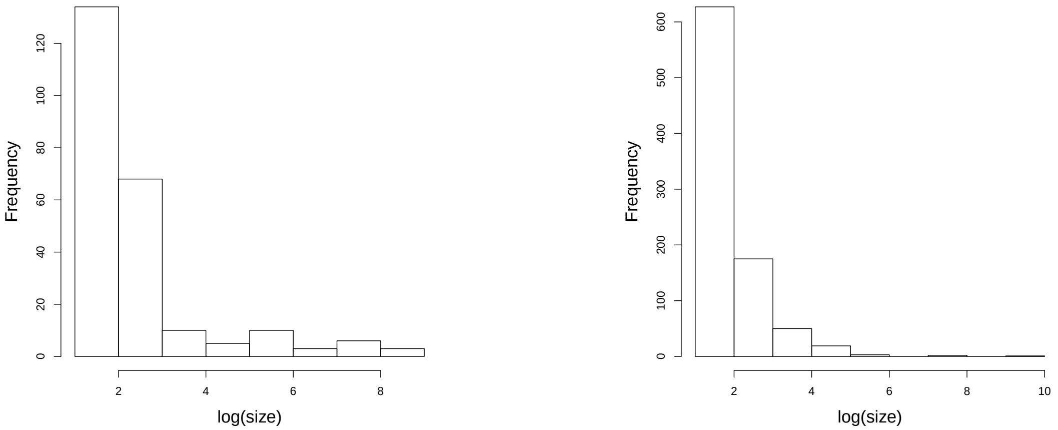
All pairs of users in the samples are connected by more than one shortest path, which justifies the necessity of path selection. Besides, a large proportion of connections are successful within limited linking steps. The best paths picked by using C_1 are mostly the same as those picked by using C_2, suggesting that the two statistics may have a positive correlation.

IV. Related work

A. Yen's algorithm

Yen's algorithm [5] computes single-source K-shortest loopless paths for a graph with non-negative edge cost. The algorithm was published by Jin Y. Yen in 1971 and employs any shortest path algorithm to find the best path, then proceeds to find $K - 1$ deviations of the best path.

Yens algorithm is known to be the efficient and widely used algorithm for determining K-shortest loopless paths [6]. Its time complexity at worst case is $O(KN_v(N_e + N_v \log N_v))$ [7].



(a)

(b)

Fig. 3. Distribution of community sizes for (a) Louvain method and (b) label propagation algorithm.

TABLE II
RESULTS WHEN LOUVAIN METHOD IS USED FOR COMMUNITY DETECTION

From	To	SP#	C_1	C_2	Success
17085	9609	7	17085 → 1029 → 2287 → 2295 → 9609	17085 → 274 → 77 → 343 → 9609	✓
12885	27890	5	12885 → 274 → 10 →8543 → 27890	12885 → 274 → 10 → 8543 → 27890	✓
32349	20592	4	32349 → 14051 → 1140 → 77 → 1589	32349 → 14051 → 1140 → 274 → 1589	✗
1764	12172	7	1764 → 96 → 12172	1764 → 96 → 12172	✓
11412	2063	5	11412 → 233 → 482 → 2063	11412 → 233 → 482 → 2063	✗
8270	25406	3	8270 → 75 → 57 → 10 → 5007	8270 → 75 → 57 → 10 → 5007	✗
7156	11071	4	7156 → 196 → 4026 →11071	7156 → 196 → 442 →11071	✓
19878	19567	4	19878 → 1769 → 152 → 4397 → 19567	19878 → 1769 → 152 → 4397 → 19567	✓
24041	16112	4	24041 → 287 → 424 → 226 → 16112	24041 → 287 → 79 → 226 → 16112	✓
12405	7156	6	12405 → 459 → 196 → 7156	12405 → 459 → 196 → 7156	✓

B. Louvain method

The Louvain method [8] for community detection is a method to extract communities from large networks created by Vincent Blondel, Jean-Loup Guillaume, Renaud Lambiotte and Etienne Lefebvre. It is today one of the most widely used method for detecting communities in large networks.

The method consists of two phases.

- Phase 1: it looks for “small” communities by optimizing modularity in a local way.
- Phase 2: it aggregates nodes of the same community and builds a new network whose nodes are the communities.
- These two phases are repeated iteratively until a maximum of modularity is attained.

The advantage of Louvain method lies in its time complexity, which is $O(N_v^2)$ for a general graph and $O(N_v)$ for a sparse graph. However, the solution depends on the order of which node or community is to be selected for evaluating the modularity value in Phase 1 since it is a greedy method.

C. Label propagation

The key idea of label propagation algorithm [9] is that each node finds a neighboring community of largest size to join until the system reaches equilibrium. Its time complexity is $O(N_v)$.

In comparison with other algorithms, such as Newman’s eigenvector method [10], label propagation has advantage in its running time, amount of a priori information needed about the network structure (no parameter is required to be known beforehand) while its disadvantage is that it produces no unique solution, but an aggregate of many solutions.

V. SUMMARY AND CONCLUSIONS

The strategy in this report combines community structure and eigenvector centrality to deal with multiple shortest paths. Two versions of community score and one importance score are defined as criteria for path selection. The assumptions made are reasonable and the method works well for the Enron email dataset.

Since there is no ground truth or related social experiments, it is difficult for us to evaluate the strategy or compare it with other methods. The upper bound for the number of users allowed to link is chosen subjectively. I choose 3 here based on my personal experience.

Friendship in real life is much more complicated. Sometimes people are not willing to share their friends for various

TABLE III
RESULTS WHEN LABEL PROPAGATION IS USED FOR COMMUNITY DETECTION

From	To	SP#	C_1	C_2	Success
17085	20387	5	17085 → 1029 → 28 → 5020 → 20387	17085 → 1029 → 28 → 5020 → 20387	✓
12885	31330	7	12885 → 274 → 10 → 5574 → 31330	12885 → 274 → 10 → 5574 → 31330	✓
32349	20591	4	32349 → 14051 → 1140 → 77 → 1589	32349 → 14051 → 1140 → 77 → 1589	✗
1764	22177	5	1764 → 214 → 823 → 22177	1764 → 293 → 3126 → 22177	✓
11412	4662	5	11412 → 233 → 79 → 102 → 4662	11412 → 233 → 79 → 102 → 4662	✓
8270	29942	4	8270 → 75 → 567 → 5025 → 29942	8270 → 75 → 567 → 5025 → 29942	✓
7156	20152	5	7156 → 1140 → 7978 → 20152	7156 → 1140 → 7978 → 20152	✓
19878	27221	4	19878 → 1769 → 1341 → 4151 → 27221	19878 → 1769 → 1341 → 4151 → 27221	✓
24041	23720	4	24041 → 287 → 648 → 3310 → 23720	24041 → 287 → 517 → 3310 → 23720	✓
12405	17544	5	12405 → 459 → 77 → 274 → 17544	12405 459 77 274 17544	✓

reasons. Moreover, friendship may change over time. A friend today may not be a friend tomorrow. Therefore, an interesting problem is how to find the best path in a dynamic network. Another interesting problem is how to integrate multiple social networks, for example, Facebook and Twitter.

The computer program for this project is written in R with igraph package. All computations for the report, including plotting, cost about 22.5 minutes on my laptop (2.5 GHz Intel Core i5, 4 GB 1600 MHz DDR3).

REFERENCES

[1] B. Klimt and Y. Yang, “Introducing the enron corpus,” 2004.

[2] [Online]. Available: https://www.cs.cmu.edu/∼./enron/

[3] Barabási and Albert-László, *Linked: How Everything is Connected to Everything Else and What It Means for Business*. New York: Plume, 2003.

[4] D. J. Watts and S. H. Strogatz, “Collective dynamics of’small-world’networks,” *Nature*, vol. 393, no. 6684, pp. 409–10, 1998.

[5] J. Y. Yen, “Finding the k shortest loopless paths in a network,” *Management Science*, vol. 17, no. 11, pp. 712–716, 1971.

[6] R. Nagubadi, “K shortest path implementation,” Ph.D. dissertation, Linköping University, 2013.

[7] E. Bouillet, G. Ellinas *et al.*, *Path Routing in Mesh Optical Networks*. John Wiley & Sons, 2006.

[8] V. D. Blondel, J. Guillaume *et al.*, “Fast unfolding of communities in large networks,” *Journal of Statistical Mechanics: Theory and Experiment*, vol. 2008, no. 10, p. P10008, 2008.

[9] U. N. Raghavan, R. Albert *et al.*, “Near linear time algorithm to detect community structures in large-scale networks,” *Phys. Rev. E*, vol. 76, p. 036106, Sep 2007.

[10] M. E. J. Newman, “Detecting community structure in networks,” *The European Physical Journal B - Condensed Matter and Complex Systems*, vol. 38, no. 2, pp. 321–330, Mar. 2004. [Online]. Available: http://dx.doi.org/10.1140/epjb/e2004-00124-y

[11] T. H. Cormen, C. E. Leiserson *et al.*, *Introduction to Algorithms*, 2nd ed. McGraw-Hill Higher Education, 2001.

[12] E. D. Kolaczyk, *Statistical Analysis of Network Data: Methods and Models*, 1st ed. Springer Publishing Company, Incorporated, 2009.

[13] J. Y. Yen, “An algorithm for finding shortest routes from all source nodes to a given destination in general networks,” *Quarterly of Applied Mathematics*, vol. 27, pp. 526–530, 1970.