Krisztina Vanyi
CSCI 5454-011, Presentation
04/20/2011

Link Prediction in Social Networks

Today's topics:

Friendship prediction exercise

What is the problem?

Practical approach (FB)

Theoretical techniques

Friendship prediction exercise

Who are more likely to be friends **Albert AND Barbara** or **Cindy AND Donald**? How did you decide?

```
Name:Albert                      Name:Barbara

Friends:Ali                      Friends:Benedict
        Beth                             Dan
        Charlie                          Gabriel
        Dan                              Michael
        Emma                             Ruth
        Richard                          David
        Gabriel                          Emma
        Ruth

School:CU Boulder                School:CU Boulder
       Alabama SU                       Francis HS
       Fire HS

Hobby:Rock climbing              Hobby:Magic tricks
      Salsa dancing                    Reading

Location:Boulder CO              Location:Longmont CO
```

```
Name:Cindy                       Name:Donald

Friends:Emma                     Friends:Barnaby
        David                            Julia
        Gabriel                          Michael
        Michel                           Jocelyn
        Jared                            Leila
        John                             Bridgette
        Ruth                             Able
        Sylvia                           Dan
        Greta
        Tom
        Benedict

School:Alabama SU                School:Francis HS
       Francis HS

Hobby:Salsa dancing              Hobby:Rock climbing
      Reading                          Salsa dancing

Location:New York                Location: New York
```

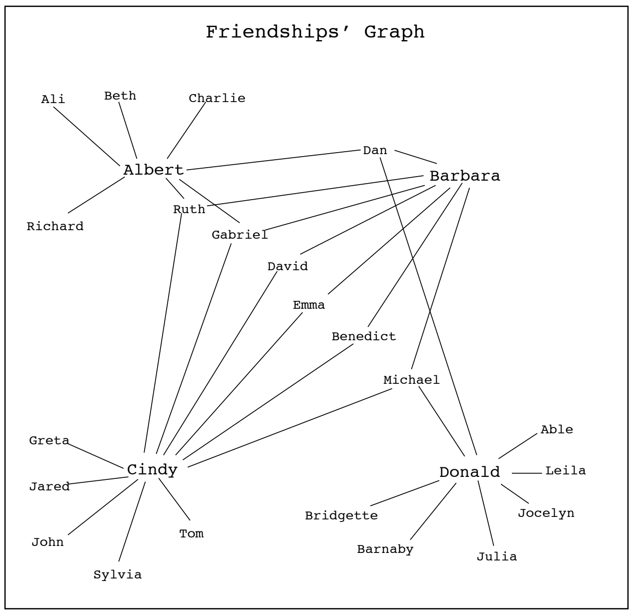
What is the problem?

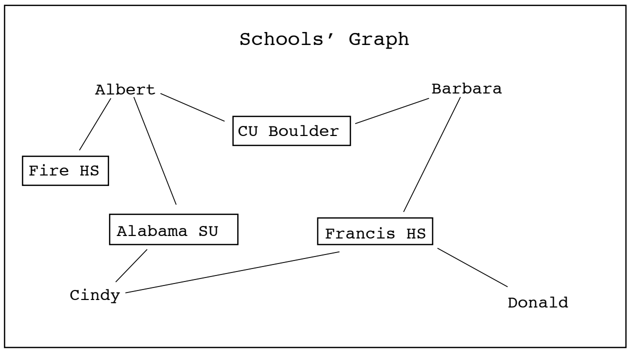
Given a static representation of a social network, we want to predict possibly existing but not yet identified connections. *Link-prediction Probelm*

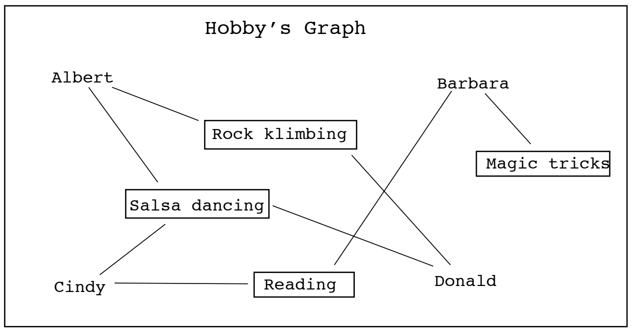
The word *predict* already tells us that we are about to face a probability problem.

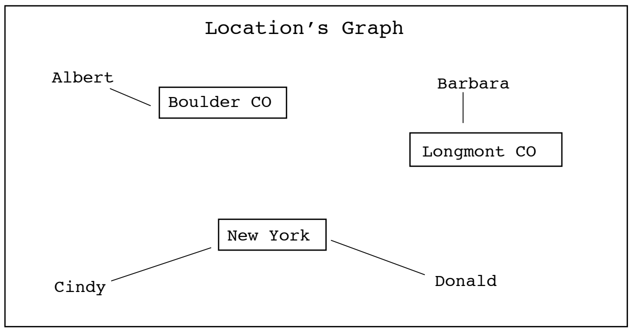
As in other such dilemmas we need a way to measure things before we can infer anything. In our case we will call this measurement ”proximity” and will use to gain a sense of distance in our network.
Talking about networks we already know that they can be represented by graphs.

For example, the connections of friends from the exercise can be represented as:









Revisiting our exercise now we ask a new question:

Out all four people (Albert, Barbara, Cindy, Donald) which pair is most likely to be friends?

How did you decide?

What we really did is very similar to what has been done in the scientific community to solve this problem.

The knowledge of topological structure of a social network can be used to predict missing connections in partly known networks with high accuracy [2]. Hierarchical decomposition can can serve as the basis for an effective method of predicting missing interactions/connections with the following techniques:

1) Given a snapshot of information about a social network we can set up the members' properties in separate subgraphs where the vertices represent our members and the weighted edges represent connections. We decide on the weights by using empirical experiments (see later in discussion of FB). Summing weights from all subgraphs for each pair of members we get the likelihood of their connection.

2) We make our predictions based on using randomly generated but equally likely models of a specific network. This can be accomplished by combining a maximum-likelyhood approach (used in statistical inference) with Monte Carlo sampling algorithm (we have heard about this in earlier lectures) [2].

Practical approach (FB)

As Lars Backstrom describes in [3] one of Facebook's services is friend suggestion. To be able to effectively infer possible unrepresented connections they use two separate methods.

First, they make ***static***, offline predictions. By observing the FB members behavior they came to the conclusion that most new friendships are established between friend-of friends (they call this distance 2 hops), and concluded that by starting with figuring out the number of common friends for a pair of people and also using other social network features (like a friend just made a new friend) they can suggest people you may know with a high probability.

They also discovered that form a practical viewpoint it is impossible to make predictions for people beyond friends-of-frineds (3 or more hops distance) because as their statistics shows the average user have 130 friends so each person is expected to have at least $130^2 \approx 17,000$ FoF. They also observed that the actual average number of FoF is even higher, about $40,000$.

As I mentioned above the first thing they look at when calculating the probability of friendship between a pair of people is the number of their mutual friends. This has been proven to be the most useful measure since statistics of actual data shows that two people with 10 mutual friends is 12 times more likely to also be friends with each other than people who have only 1 mutual friend.

After considering other properties of the members in question they combine the probabilities to establish a final value based on the combination of mutual friend number and length of friendships.

$$v(FoF) = \sum_{f_i} \frac{(\delta_{u,f_i} \cdot \delta_{f_i,FoF})^{-0.3}}{\sqrt{friends_{f_i}}}$$

where u is the source person, $\delta_{\_,\_}$ is the time since friendship made, FoF is the person with whom we try to establish the probability of a friendship and $friends_{f_i}$ is the number of mutual friends with respect to the particular direct friend.

Even though this one combined value is already performs above random friendship prediction FB examines more personal properties such as weighted (how close - lots of/little communication) mutual friendships and demographic data, like country, FB age of source user and friend count.

In practical terms (time and machine requirements) suggesting new friends is "expensive". As I mentioned before FB statistics shows that the average number of FoF is about $40,000$ for each member. Furthermore we also need to consider that FB has over $500M$ users. Which gives us $40K \cdot 500M = 20Trillion$.

Handling such a huge number of required friend predictions they had to apply an executable method.

First they use a heuristic method to make an initial elimination of possible suggestions.

Then by a logistic regression they select 1000 FoF for a particular source user. (Linear time rank-N algorithm)

Next they run full decision tree algorithm only these selected FoF's.

After that they select top 100 out of the fully ranked 1000. These are the eligible candidates for showing it to source user.

Since this process takes a lot of time and machine capacity (in spite of ranking $145,000,000$ suggestions per second) FB only runs the static suggestion process in every 2 days. On the other hand, to encourage new user friend expansion they run it more often for them.

Since this would provide a fairly static prediction FB also applies a more ***dynamic*** approach to revise its suggestions.

They provide each source user the opportunity to delete people from he suggestion list and these FoF's will not be evaluated the next time around so new ones can be presented.

Another consideration is the number of times a source user was shown the suggestion. Generally after 4 showing the friend candidate is viewed as uninteresting for the source user so it is also eliminated from the list.

To further improve their friend suggestion success rate (number of new friendships made) FB also considers the following source user behavior:

Number of friends added based on FB suggestion.

Number of suggested friends rejected.

This dynamic approach is also called "training" and by applying this technique for each user (customization) FB has been able to hugely increase its friend prediction performance.

Theoretical techniques

As described in [1] and [2] link prediction can be used in other than FB type social networks as well. The examples we can see in these papers are:

metabolic network of the spirochaete (http://mw2.m-w.com/medical/spirochaetes),

- *Treponema pallidum*, a network of association between terrorists,
- food web of grassland spices,
- scientists' collaboration network (Erdős number)

In theoretical approaches modeling takes the main stage since rather working with a static snapshot of a network it is interested in network evolution and uses techniques that tends to take into account specific attributes of the nodes in the network, rather than evaluating the power of prediction methods that are based purely on the graph structure [1].

An example of this method follows:

Given 5 coauthor networks pick one. On this particular one they establish a 3-year training and a 3-year test interval. During the training interval they observe cooperation between the network members and also the production level (number of papers published). During the following test interval new cooperations (formerly did not work together) are noted and selected for examination. Having all this data not we need to see if we can predict this new interactions from creating models from the data acquired during the training period.

In practical terms it means that we know what new connections were made and wan to find a method based on this "experiment" that can predict the probability of links in the future.

A number of methods have been developed to solve this problem and all of them have some basic common characteristic.

- All of them assign a weight to pairs of nodes (members in the network).
- They are based on an input graph (the chosen one from the 5 possible in the example of coauthors).
- In each model a ranked list is produced in decreasing order of weights of pair of nodes.
- All of them examines all possible generated network models to make predictions.

Because these similarities the solution methods can be viewed as computation of a measure of proximity/similarity.[1]

The **first** method we want to introduce is based on node neighborhoods.

This approach is based on the hypothesis that two nodes, (x, y), are more likely to form a link in the future if their sets of neighbors $\Gamma(x)$, $\Gamma(y)$ have a large overlap [1] (This already sounds familiar, isn't it?)

Now we have the choice of score (probability) assignment.

- *Common neighbors*: $score(x, y) = |\Gamma(x) \cap \Gamma(y)|$

- *Jaccard's coefficient and Adamic/Adar*: $score(x, y) = |\Gamma(x) \cap \Gamma(y)| \ / \ |\Gamma(x) \cup \Gamma(y)|$

- *Preferential attachment*: $score(x, y) = |\Gamma(x)| \cdot |\Gamma(y)|$

The **second** method is based on the ensemble of all paths. We again have a choice of scoring.

- *Katz*: $score(x, y) = \sum_{l-1}^{\infty} \beta^l \cdot |paths_{x,y}^{<l>}|$

where $paths_{x,y}^{<l>}$ is the set of all length-l paths from x to y, and $\beta > 0$ is a parameter of the predictor.

- *Hitting time, Page Rank, and variants*:

Let $H_{x,y}$ be the *hitting time* (expected number of steps) of y starting a random walk from x. Since the hitting time is usually not symmetric we also consider the *cummute time*, $C_{x,y} = H_{x,y} + H_{y,x}$. Since both of these measures serve as natural proximity measures, they can be used as $score(x, y)$.

To compensate for the fact that the hitting time is small when the target node has a large stationary probability, π_y, a modified possible score is

$score(x, y) = -H_{x,y} \cdot \pi_y = -(H_{x,y} \cdot \pi_y + H_{y,x} \cdot \pi_x)$

To avoid the next possible problem, these measures also depend on parts of the graph far away from source and target nodes, we need to allow the random walks to reset time after time. This way far away graph parts will not be explored and therefore will not interfere with our predictions. Such random resets are also a basis for PageRank measures for Web-pages.

- *Sim rank*:

Here we define similarity: two nodes are similar to the extent that they are joined to similar neighbors. By definition $similarity(x, x) = 1$ and:

$$similarity(x, y) = \gamma \cdot \frac{\sum_{a \in \Gamma(x)} similarity(a, b)}{|\Gamma(x) \cdot \Gamma(y)|}$$

Where $\gamma \in [0, 1]$ is a parameter.

The **third** method is a higher-level approach. These can be used together with any of the formerly listed methods to improve them.

- *Low-rank approximation*: If we use the common-neighbour method we can modify it by letting $r(x)$ and $r(y)$ be rows in the matrix M that represent the neighbors of x and y respectively and then take the inner product (dot product) of $r(x)$ and $r(y)$ in order to define $score(x, y)$.

- *Unseen bigrams*: this method closely relates to language modeling since it is concerned with estimating frequencies of pairs of words co-occuring in a text.

In practical terms, we first need to compute scores by one of the above methods then let $S_x^{<\delta>}$ denote the δ nodes that are most similar to x, where $\delta \in \mathbb{Z}^+$.

Now we are ready to define enhanced scores:

$$score^*_{unweighted}(x, y) = |\{z : zin\Gamma(y) \cap S_x^{<\delta>}\}|$$

$$score^*_{weighted}(x, y) = \sum_{z \in \Gamma(y) \cap S_x^{<\delta>}} score(x, z).$$

- *Clustering*: This method is based on deleting "less relevant" edges in order to clean up our initial graph and therefore be able to make more relevant predictions with our choice of main method.

Considering all these possibilities there is still plenty of opportunity to improve the theoretical approach since according [1] even the best (*Katz*) theoretical prediction method is correct only in 16% total.

Bibliography

[1] David Liben-Nowell and John Kleinberg, The Link-Prediction Problem for Social Networks.

[2] Aaron Clauset, Christopher Moore and M. E. J. Newman, Hierarchical structure and the prediction of missing links in networks.

[3] Lars Backstrom, People you may know 2010.