Analyzing Social Network Data

Name: Yuki Kitayama

Date: 2020-09-02

Overview

This project researches information flow in a social network. The first part of the project will explore friendship in a Facebook data and suggestion the potential new connections. In the second part, the goal will be to construct information flow in the network of UC San Diego Facebook data, and to identify how a new technology diffuse given a particular condition.

Data

The provided UCSD Facebook data. This network contains 14,948 people, or vertices, and the average number of friends that each people has, or edges, is 59.3. The data contains 2 integers separated by space in each line representing the friendship between 2 people, and the same connection exists in an opposite manner. For example, line 1: 0 1, and line 2: 1 0, meaning that a person 0 and a person 1 are friends with each other.

Questions

Easier

For a given user, which of their friends are not connected as friends, and can we suggest a pair of two people as a potential new friendship? For example, if the given user, Yuki, is friends with both Danny and Tom, and if Danny and Tom are not friends, we will suggest them as potential friends.

Harder

Suppose that people in this network use an old technology B, and a set of people start to use a new better technology A. How does this new technology disseminate in the network? How can we model this information spread, or cascade? On what conditions, can a new technology spread or cannot? If the cascade occurs, does it completely dominate the entire network, or does the coexistence happen with the certain proportion of users, technology A or B?

Algorithms and Data Structures

Easier Question

The main data structure is that the network has been laid out as a graph with an adjacency list (https://en.wikipedia.org/wiki/Adjacency_list). Each individual in the graph is a vertex and it has edges between vertices representing a friendship.

Algorithm is that, input is a specific user (U), and output is a list of pairs of unconnected potential friends. All the vertices are integers in the data, so if we input an integer value to this method, it outputs an ArrayList of java.util.Pair of two integers. First, get a set of friends of U by exploring all the edges of U. Initialize an empty set to contain integers, which keeps track of which keys are added to the list to avoid redundant duplicated pairs in the result. For example, it produces only Pair<0, 1> in the list, not Pair<0, 1> and Pair<1, 0>. Then create a returned list of pairs of vertices, as followed,

Algorithm

```
Initialize an empty set of pairs of integers,
For each friend X in the list,
  For each friend Y in the list,
    If (X and Y are not the same && X is not already friends with Y && the set does not contain Y)
      Add pair (X, Y) to the returned list
      Add X to the set
Return the list.
```

There could be no new connection suggestion, it could be huge in the UCSD Facebook data so that the returned list is conditioned as followed,

- If suggestion is 0, print there is no friend suggestion
- If the number of suggestions is less than or equal to 5, just return the list
- If the number of suggestions is greater than 5, randomly choose 5 pairs from the list and return the sampled list.

Use a hash set to store edges, and to find whether X and Y are friends, and O(1), and we have two for loops so O(V) and O(V), so in the end, asymptotic runtime is O(V^2). It is followed by if statements, and the asymptotic runtime is O(1).

Data structure

We make a Java Class which implements Interface Graph. This Class reads Facebook data, contains network with vertices and edges, and has many methods to operate on the social network. We set this friends suggestion as one of the methods.

Harder Question

The diffusion of innovation from a net technology is modeled by the idea of cascade from a book, "Networks, Crowds, and Markets" by David Easley and Jon Kleinberg, Chapter 19. We use 3 hyperparameters; a set of initial adopters of a new technology A, payoff A of using the technology A, and payoff B which assumes that people other than the initial adopters in the network use an old technology B. We calculate P, proportion of neighbors which use a technology A. Then, we can model cascade by making the following equation,

$$P \geq \frac{B}{A+B}$$

It means that the proportion of people using a new technology A is greater than the payoff that you get by using an old technology B. So, this model considers both how your neighbors are reacting to a new technology and what the intrinsic value of a new technology gives you. If P, called cascade capacity, is large, cascades happen more easily.

The output of this algorithm is dependent on those hyperparameters so that we experiment the grid of combinations of different parameters. From the right-hand side of the equation, relative increase of A to B matters so that B is fixed at 1, and only change A. For example, A = 2, and B = 1 means that A's payoff is twice as big as B’s payoff. P is determined by the number of the initial adopters and their distribution in the network. We randomly choose initial adopters from the network, and use the same seed for every experiment.

Complete cascade is defined as the state that everyone in the set of people who use technology B are converted into the users of technology B by the initial adopters, payoff A and the iteration. Complete cascade is a Boolean result, but we calculate cascade ratio by number of people who use technology A divided by the number of people in the network after the iteration. With these, we can answer to the question how we can get a complete cascade.

Algorithm

In cascading, a certain vertex is converted from B to A, and this conversion affects P of the other vertices. So, the cascading algorithm is iteratively performed to the network until no change occurs.

```
Initialize a set of initial adopters with size n and picked randomly with seed.
Set payoff A and payoff B
Set set A from initial adopters set
Calculate threshold by B divided by the sum of A and B
Initialize Boolean switching = true
While (switching)
  Initialize hash set of integers
  Initialize integer of counter
  Get iterator of set A
  While (iterator.hasNext())
    Get neighbors of a vertex from set A
    For (neighbor : neighbors)
      Initialize counterA to count number of people using A
      Get neighbors of neighbor
      Get size of the neighbors d
      For each neighbors
        If set A contains this neighbor, plus 1 to counterA
      Calculate P by counterA divided by d
      If p > threshold && set A does not contain neighbor && candidate does not contain neighbor
        Add neighbor to candidates set
        Plus 1 to counter
  Add all candidates to set A
  If counter is 0
    Set false to switching
```

Data structure

Initial adopters are set by setInitialAdopters method outside of this cascade method, which allows us to try different sizes of initial adopter randomly. This cascade method also has parameters of payoff A and B. So, we can try different combinations of initial adopters, payoff A, and payoff B to experiment how a cascade occurs. A result is the following.

The values in a table below are cascade ratios. Initial adopters are randomly chosen with the seed fixed. As a result, it was hard to find a nice payoff A or initial adopters that produces a nice coexistence of technology A and B. The outcome is rather all or nothing, meaning very low cascade ratio, or almost complete cascade ratio.

Payoff A \ Initial adopters	1	10	100	250	500	1000
1	0.000	0.001	0.007	0.017	0.034	0.067
1.5	0.000	0.001	0.007	0.017	0.035	0.070
2	0.000	0.001	0.007	0.017	0.035	0.070
2.5	0.000	0.001	0.007	0.018	0.037	0.076
3	0.000	0.001	0.008	0.018	0.038	0.080
5	0.000	0.001	0.009	0.026	0.999	0.999
7.5	0.000	0.001	0.999	0.999	0.999	0.999
10	0.000	0.001	0.999	0.999	0.999	0.999

Algorithm Analysis, Limitations, Risk

Easier Question

When the network is huge, friend suggestion might be a very long list of pairs of potential connections. So, in the method, I implemented maximum number of suggestions is 5. If suggestion is bigger than 5, randomly pick 5 pairs from the all the potential connections.

Harder Question

This cascade assumes that we only have two choice, becoming A or B. But in practice, it is not realistic because there should be a bilingual type of people. It means a person originally use an old technology B observes that the neighbor started to use A or payoff looks sufficiently bigger than B, and maybe just starts trial of A, not an instant complete conversion from B to A, and just use both A and B. This can be modeled by introducing another parameter C, a cost of being bilingual. This model is described in the chapter 19 of Networks, Crowds, and Markets textbook, but this project does not implement this model.

Other Risk

Harder question intends to answer when a complete cascade occurs. However, depending on the network structure, it is impossible to occur, not because of the cascade method mistake. When there is an isolated network, and when they are not chosen as initial adopters, a complete cascade fails to occur. The following is the list of vertices found in the UCSD Facebook data. There are vertices in the Facebook UCSD network which are pairs of vertices where one is only connected to the other, each having only one friendship, and we cannot calculate P, a proportion of neighbors using a new technology A.

Isolated vertices: [1938, 4171, 4950, 7022, 8109, 10102, 14070, 14697]

Their network: 1938=14070, 4171=14697, 4950=10102, 7022=8109.