

Brady Cassada

Algorithms

Lab 5: Graphs

Source Code: <https://github.com/cassadab/graphs-lab>

Problem 1: Friend Recommendation System

In the source code you will find two recommendation tests. One with a maximum length of two and another with a maximum length of five. Additionally, there are a few unit tests that cover the breadth first algorithm and the recommendation algorithm.

Below is a table of the data collected from the two tests:

Max Length	True Pos.	False Pos.	False Neg.	Precision	Recall
2	2740	150648	66	0.018	0.977
5	2802	557232	4	0.005	0.999

A max length of two essentially considers “friends of a friend” or what LinkedIn likes to call a 2nd level connection. Increasing this number considers many more people who are much less connected to the original person. With this in mind, it makes sense that we see a large jump in false positives (a predicted connection/edge that was not found in the testing set) from the first test to the second. A low precision also makes sense considering the algorithm is finding every possible connection. It would be unreasonable to expect all of these to be found in the test data.

However, both tests have a very high recall. This is because of the low number of false negatives. Coming in at about 98% and almost 100%, just about all of the edges from the test data were found in the predicted set.

Problem 2: Determine Order of Task Execution

As you can see, this section was coded in Python. I was getting a bit bored with Java and I realized I’d have to re-structure a lot of my code to fit it to problem 2 anyway.

The output of the tool given the shrimp.mf file starts on the next page:

[illegible]

[illegible]

