

We have chosen to use the Ego-Twitter data set from the Stanford Data Sets. This data set represents social circles with nodes representing profiles and directed edges representing followers. If this does not work on EWS, then we plan to use the soc-LiveJournal1 dataset, also from Stanford Data Sets. Similarly to Twitter, the nodes represent users and the edges represent friends, decided by one user.

We decided to use a BFS traversal, as it will be quicker to find people who are more closely connected to each other. For example, if we are searching for one person, starting from one of their close friends (who follow each other), we should want to find them quickly. If we use DFS, it could take going through nearly every user to find the friend, however with BFS the friend would be in the first level outside of the node, and would be found much quicker.

We are choosing to implement Dijkstra's Algorithm to find the shortest path between two accounts, or even one account and all other accounts. We want to sort of test the idea of "six degrees of separation", by seeing how short the path is from a random user to all other users. Dijkstra's Algorithm does exactly this, hence why we chose it.

We also want to implement the PageRank algorithm on the data. This will show which accounts are most popular relative to the other profiles. PageRank does not just look at how many edges point to a node, but also how many edges the nodes that point to the main node have pointing to them, and so on. This would allow us to see what profiles have the most "clout", as they might have the most important followers.