Introduction

This project consisted of 4 steps.

1. Crawl a social networking site for datasets
2. Analyze them through graph measures such as bridges, diameter etc.
3. Analyze the importance of each node using centrality, page rank etc.
4. Simulate Random graph, small world model, and preferential attachment model for the crawled graph and see how the simulation matches up with the graph obtained from crawling.

Part – 1 – Crawl for Datasets

The social networking site chosen was Netlog which has now changed it's name to Twoo.com. This is a dating site. Hence, the datasets weren't easily available. It wasn't possible to crawl from user to user and establish links.

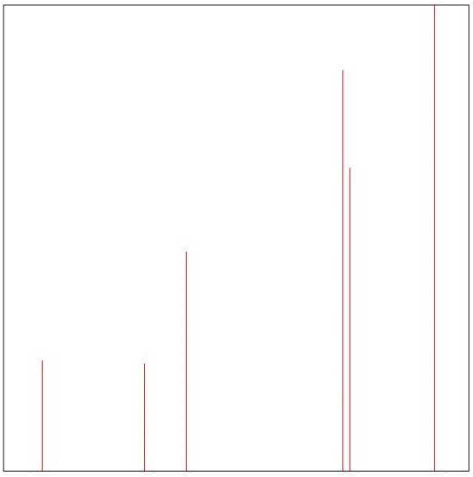
However, there was a search page which was populated with 50 common American names from ages 18 to 50 within 15 miles of Surprise, Glendale, Tempe and Chandler each. The results obtained were put in 4 separate regions. There were some common results between overlapping areas such as Tempe and Chandler, Tempe and Glendale and Glendale and Surprise. In this case, there was a little bit of post processing that was done to remove duplicates.

Since, there was no social interaction that was provided by crawling itself, the user sets obtained from each region were modelled as connected to each other. The common ones found between each region were modelled as connected between both regions. Since no node produced a very large, no sampling was done.

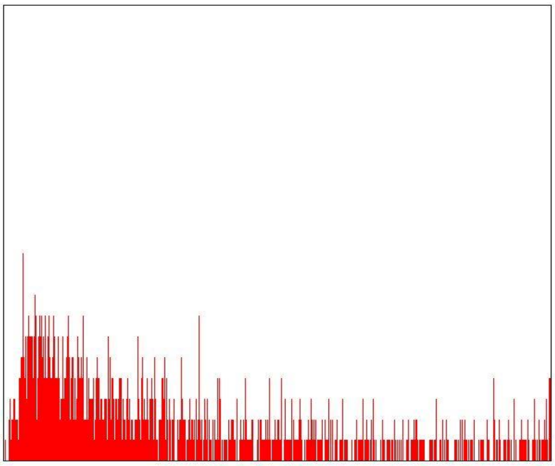
In order to facilitate crawling the library scrapy was used. Python was the language for crawling and in fact for all parts of the project. Results are available in the submission package.

Part – 2 – Graph Essentials

As explained in the previous section, since the connections weren't modelled on true social interactions, the degree distribution was obtained as shown on the figure on the left side. However, on reconnecting the edges to obtain a power law distribution, the figure on the right side was how the degree distribution looked. The figure on the right is the degree distribution obtained with exponent 2.



Diameter of this graph is 5
Number of bridges are 0



Diameter of this graph is 2
Number of Bridges are 0

On removing x% of the edges in the graph, the largest connected component stayed at 1517 until 99%

The data was generated using the igraph library available with a python interface.

Part – 3 – Network Measures

The following are the statistics of the network that was crawled.

Average Local Clustering Coefficient	0.681417154172
Global Clustering Coefficient	0.603163432474
Average path Length	1.52354146411

Following are the top 10 page ranks, Eigen Vector Centralities and Degree Centrality

Page Rank	Eigen Vector Centrality	Degree Centrality
0.00129998162277	1.0	1516
0.00129998162277	1.0	1516
0.00129998162277	1.0	1516
0.00129998162277	1.0	1516
0.00129998162277	1.0	1516
0.00129998162277	1.0	1516
0.00129998162277	1.0	1516

Following are the rank correlations between each of them

Page Rank and Eigen Vector	0.99762144943657682
Page Rank and Degree	0.9999575862288727
Degree and Eigen Vector	0.99820474171141627

Highest Jaccard Similarity is **0.998681608438**

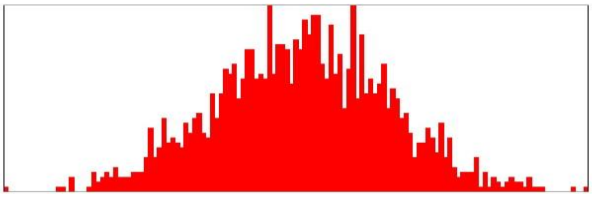
Because the crawl produced datasets that were very similar, you would see that the rankings are pretty similar, similarity is high. Hence, each node is very similarly connected to each other.

Part – 4 Network Models

This section compares the Random Graph, Small World and Preferential attachment models with the graph generated from the crawl on the basis of average path length, global clustering coefficient and degree distribution. All graphs were obtained by using the igraph library.

Property	Crawled Graph	Random Graph	Small World Model	Preferential Attachment
Global Clustering Coefficient	0.603163432474	0.476490486887	0.5	1.0
Average Path Length	1.52354146411	1.52354146411	190	1.0
Degree Distributions	The degree distributions are shown below			

Random Graph – Degree Distribution



The small world and preferential attachment degree distributions weren't available at the time the report was made.