SOCIAL NETWORKS PROJECT REPORT

Determining Important Connections in Social Network using Page Rank Algorithm

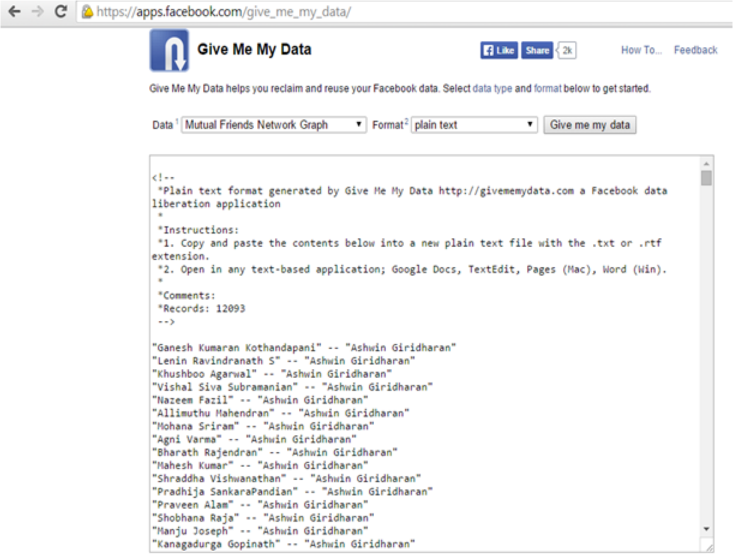
Ashwin Giridharan
Vineeth Lakshminarayanan
May 19, 2015

1 INTRODUCTION

With the increase in priorities given to a social network connection in terms of trust and opportunities, it may be useful to determine the importance of a connection in one's social network. The importance of a connection "C" can be derived using the naive approach of taking into account the number of connections to C in the network. But intuitively it makes sense that the quality of C's connections matters rather than the quantity. Hence we propose to use Google's PageRank algorithm to rank connections in the social network, thus taking into account both the number and quality of a contact's connections. We rely on Google's PageRank assumption that more important connections tend to have more quality connections within the network.
The ranking of connections from PageRank algorithm is compared with the ranking of based on graph "In-degree" to analyze if ranking by PageRank makes more sense in predicting quality connections.We used two variations of PageRank algorithm, Power Method and Linear System Formulation.
Work Done:
Ashwin Giridharan - Power Method
Vineeth Lakshminarayanan - Linear System Formulation

2 DATA

The Facebook mutual friends graph data of one of the author "Ashwin Giridharan", will be referred as subject throughout the paper, with 510 nodes/connections and 12093 edges/relation is used for the experimentation. The graph data is obtained from Facebook App "Give Me My Data". (Note that you can no longer get your complete information from this app, as Facebook introduced restricted policy in sharing public data from May 01 2015).



Give Me My Data helps you reclaim and reuse your Facebook data. Select data type and format below to get started.

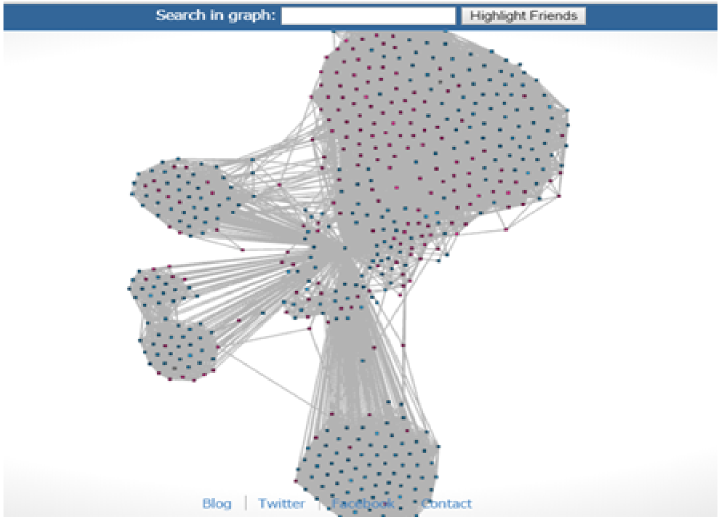
Data [1] Mutual Friends Network Graph ▾ Format [2] plain text ▾ Give me my data

```
<!--
 *Plain text format generated by Give Me My Data http://givememydata.com a Facebook data
liberation application
 *
 *Instructions:
 *1. Copy and paste the contents below into a new plain text file with the .txt or .rtf
extension.
 *2. Open in any text-based application; Google Docs, TextEdit, Pages (Mac), Word (Win).
 *
 *Comments:
 *Records: 12093
 -->

"Ganesh Kumaran Kothandapani" -- "Ashwin Giridharan"
"Lenin Ravindranath S" -- "Ashwin Giridharan"
"Khushboo Agarwal" -- "Ashwin Giridharan"
"Vishal Siva Subramanian" -- "Ashwin Giridharan"
"Nazeem Fazil" -- "Ashwin Giridharan"
"Allimuthu Mahendran" -- "Ashwin Giridharan"
"Mohana Sriram" -- "Ashwin Giridharan"
"Agni Varma" -- "Ashwin Giridharan"
"Bharath Rajendran" -- "Ashwin Giridharan"
"Mahesh Kumar" -- "Ashwin Giridharan"
"Shraddha Vishwanathan" -- "Ashwin Giridharan"
"Pradhija SankaraPandian" -- "Ashwin Giridharan"
"Praveen Alam" -- "Ashwin Giridharan"
"Shobhana Raja" -- "Ashwin Giridharan"
"Manju Joseph" -- "Ashwin Giridharan"
"Kanagadurga Gopinath" -- "Ashwin Giridharan"
```

3 USE OF CLUSTER VISUALIZATION

The interactive cluster visualization of the friends network data from 'yasiv.com/facebook', helped in analyzing why 'Page Rank algorithm' would rank certain connections higher and certain connections lower. Many interesting facts were deduced by using this interactive visualization and the final ranking from Page Rank, which are reported in the final section. (Note that you can no longer get your complete information from this app, as Facebook introduced restricted policy in sharing public data from May 01 2015).



4 RANKING CONNECTIONS USING PAGE RANK - POWER METHOD

The connections of the subject in this approach are ranked using Google's PageRank algorithm. The following steps were involved in determining the ranking vector:

1. Constructed 510 * 510 adjacency matrix.

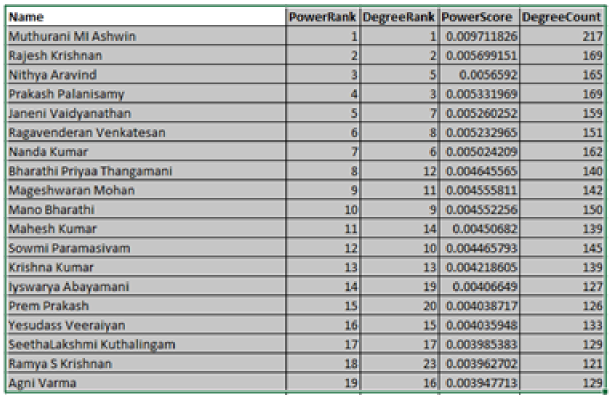
2. Converted adjacency matrix to probability matrix, to make it stochastic.
3. Assigned minimum probability to handle sparse rows, to make it irreducible.
4. Applied Power Method to generate Stationary/Rank Vector with scaling factor as 0.99 and tolerance level as 10^{-10}.
   (a) Equation for Power Method is $\pi^T(\bar{\bar{P}}) = \pi^T$ where $\bar{\bar{P}} = \alpha\bar{P} + (1-\alpha)ee^T/n$. Here $0 <= \alpha <= 1$ and $E = 1/nee^T$. This convex combination of the stochastic matrix $\bar{P}$ and a stochastic perturbation matrix E insures that $\bar{\bar{P}}$ is both stochastic and irreducible. The irreducibility adjustment also insures that P is primitive, which implies that the power method will converge to the stationary PageRank vector π^T.
5. Generated map rank for connections from Rank vector

5 Ranking connections using Graph In-Degree

The connections in this approach were ranked in the order of number of mutual friends between the subject and his connections or in other words taking into consideration only the in-degree of a node.

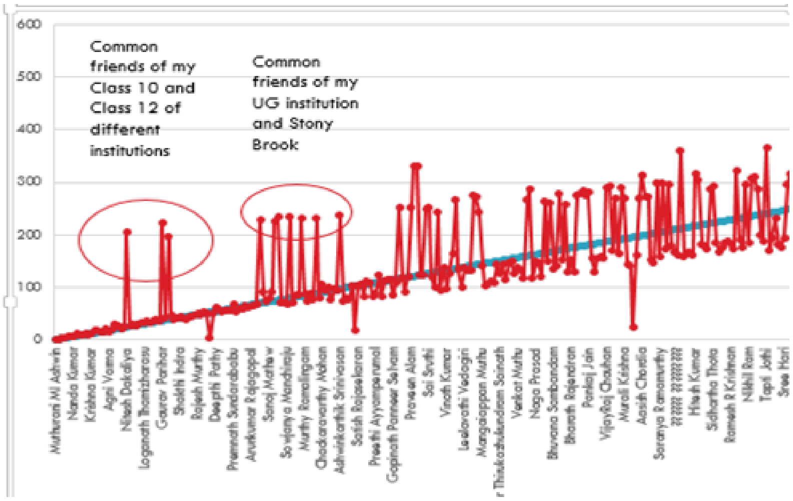
6 Analyzing the ranking by Power Method and In-Degree Method

The ranks for connections generated by the respective approaches were compared to analyze if Page Rank makes more sense in ranking the subject's connections by considering its influence across the entire network.



Name	PowerRank	DegreeRank	PowerScore	DegreeCount
Muthurani MI Ashwin	1	1	0.009711826	217
Rajesh Krishnan	2	2	0.005699151	169
Nithya Aravind	3	5	0.0056592	165
Prakash Palanisamy	4	3	0.005331969	169
Janeni Vaidyanathan	5	7	0.005260252	159
Ragavenderan Venkatesan	6	8	0.005232965	151
Nanda Kumar	7	6	0.005024209	162
Bharathi Priyaa Thangamani	8	12	0.004645565	140
Mageshwaran Mohan	9	11	0.004555811	142
Mano Bharathi	10	9	0.004552256	150
Mahesh Kumar	11	14	0.00450682	139
Sowmi Paramasivam	12	10	0.004465793	145
Krishna Kumar	13	13	0.004218605	139
Iyswarya Abayamani	14	19	0.00406649	127
Prem Prakash	15	20	0.004038717	126
Yesudass Veeraiyan	16	15	0.004035948	133
SeethaLakshmi Kuthalingam	17	17	0.003985383	129
Ramya S Krishnan	18	23	0.003962702	121
Agni Varma	19	16	0.003947713	129

It is indeed verified that PageRank assigns better rank to connections which are more influential in the subject's network or being a favorite connection among the subject's connections, rather than mere measurement of total mutual friends between the subject and this connection.



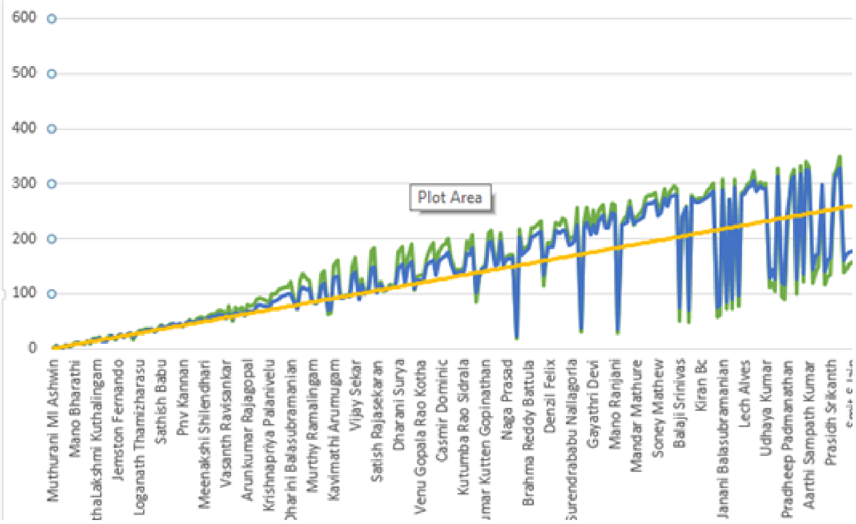
Here in this figure, the blue line represents the ranking by Page Rank Power method and the red line indicates the ranking by in-degree method. Top 250 connections ranked by Power method displayed in the graph.
As the circle markers in the figure illustrates, the connections ranked lower by in-degree method were ranked higher by Page Rank method, due to these connections establishing heavy bonding with other influential connections in subject's network and they being connected to major groups of the subject.

7 Tinkering with Scaling factor and Tolerance Level for Power Method

The rank vector for connections using PageRank algorithm was generated with different combinations of scaling factor and tolerance factor to determine the convergence rate in arriving at the stationary rank vector.

Convergence Rate (Iterations)		Scaling Factor					
		0.75	0.8	0.85	0.9	0.95	0.99
Tolerance Factor	10 ^ -6	12	15	19	26	40	68
	10 ^ -8	25	31	41	56	86	164
	10 ^ -10	39	48	62	86	138	280

It is observed that the convergence rate decreases with increase in scaling and tolerance factor. Also the ranking vectors generate by extremes of scaling and tolerance factor is compared to understand the role of these factors in determining the influential connection.



Here in this figure, the blue line represents the ranking with scaling factor: 0.75 and tolerance: 10^{-6}, the green line indicates the ranking with scaling factor: 0.85 and tolerance: 10^{-8} and the yellow line indicates the ranking with scaling factor: 0.99 and tolerance: 10^{-10}. Here, Top 250 connections ranked by Power method with scaling factor: 0.99 and tolerance: 10^{-10}, are displayed in the graph.

It is observed that many connections which are ranked in the ranges of 200 by lower scaling and tolerance factors are ranked within 100 by the higher scaling and tolerance factor. Also the connections that are ranked between 50 and 100 with the lower scaling and tolerance factors are ranked above 250 with higher factors. This reveals the extreme to which the scaling and tolerance factors determines the accuracy of ranking influential connections. The factors best determining the influential connection is described in "Validation by subject's experience and opinion" section.

8 Validations for PageRank - Power Method

It is observed that the Page Rank Power Method with higher the value of scaling and tolerance, better the ranking of influential connections. This conclusion was arrived based on the following validations.

8.1 Validation by introducing dummy/spam connections

Several spam connections are injected in such a way they have more mutual connections with the subject and these mutual connections are in turn dummies, sharing relationship only with the subject. Dummy connections with dummy mutual friends count between 300 and 500 are injected and it is observed that these dummy connections fall between 85^{th} and 95^{th} percentile in the overall ranking by Power method against being placed in top 5 percentile in ranking by in-degree.

8.2 Validation by subject's experience and opinion

Since the friend's network used belongs to the author of this report, this validation is done by assessing the amount of quality time the author/subject spent with these connections. This assessment agrees more with the results of ranking with scaling factor: 0.99 and tolerance: 10^{-10}

8.3 Validation using the Interactive Visualization

This validation is done using the interactive visualization from "yasiv.com/facebook", where it was evident that the connections "CList" of the top 20 ranked connections by PageRank, are centered in their respective clusters showing their strong influence in their connecting groups and overall network. Further 60% of the connections in CList fall in the top 20 percentile of the rank vector.

9 Conclusion for PageRank - Power Method

It is evident from the above observations and validations that Page Rank makes more sense in predicting influential connections in friend network and the predictions are even more accurate when scaling factor and tolerance level are increased. Here the scaling factor of 0.99 and the tolerance factor of 10^{-10} fits better to this model than Google's standard scaling factor of 0.85 being used in their web page ranking, as 0.99 reflects the actual structure of the network graph and there is no significant change is observed in ranking beyond the tolerance level of 10^{-10}.

10 Ranking connections using Page Rank - Linear System Formulation

Traditionally, PageRank has been computed as the principle eigen vector of a Markov chain probability transition matrix. The transition probability matrix (**P** matrix) is constructed and the power method is used to converge to the stationary PageRank vector. The rate of convergence highly depends on the size of the transition probability matrix. If the size of the probability matrix is very large, then the power method will take a large number of iterations to converge to the stationary PageRank vector. All the changes to the power method assumes the Markov chain and solves it.

The Linear System Formulation is a technique where we formulate the PageRank problem as a linear system. Formulating the PageRank problem as a linear system makes solving the PageRank problem very easy without any iterations. The Linear System Formulation makes use of the fact that there will be many rows in the transition probability matrix whose values will all be zeros. These rows correspond to dangling nodes. Dangling nodes refers to those nodes that do not have any outgoing links at all. In case of the Internet, these refer to jpeg images, png images and websites that do not have any hyperlinks.

The intuition behind the Linear System Formulation is that the **P** matrix consists of a large number of zero rows (dangling nodes). These rows can be permuted and shifted to the bottom so that the transition probability matrix is reduced, which in turn reduces the computation required, thereby increasing the computational speed. The eigen vector problem is given as follows:

$$\pi^T(\alpha\bar{P} + (1-\alpha)ev^T) = \pi^T$$

When the above equation is rewritten, we get,

$$\pi^T(I - \alpha\bar{P}) = (1-\alpha)v^T$$

We know that $\pi^T e = 1$. This is known as the normalizing constant.The various steps in Linear System Formulation are as given in the following paragraph.

The rows and columns of matrix **P** (transition probability matrix) are permuted so that the rows corresponding to the dangling nodes are shifted to the bottom of the matrix **P**.

We solve for π_1^T **in** $\pi_1^T(I - \alpha P_{11}) = v_1^T$**.** Here v_1^T corresponds to the non-dangling section of the nodes whereas v_2^T corresponds to the dangling portion of the nodes.

We compute $\pi_2^T = \alpha\pi_1^T P_{12} + v_1^T$**.**

Finally we normalize the value $[\pi_1^T \pi_2^T]/\|\pi_1^T \pi_2^T\|$

10.1 Data Set

Details of the data set are given as follows:

Source of Data Set: https://apps.facebook.com/give_me_my_data/

Size of transition probability matrix, P: 550 * 550 (including dangling nodes)

Method of Probability Calculation: If a node n_i consists of k links, then each and every value in the adjacency matrix is replaced by $1/k$ for the corresponding row.
Value of α: 0.85
Value of personalization vector, v: 0.002, 0.002, 0.002 ...
After the transition probability matrix was constructed, the above algorithm was implemented in order to calculate the stationary PageRank vector. The size of the P_{11} and P_{12} matrices were 510 * 510 and 510 * 40 respectively. Thus we have exploited the existence of dangling nodes to reduce the computation of the PageRank vector. In our case, we have reduced the **P** matrix from 550 *550 to a 510 * 510 matrix.
Once the algorithm is run, we analyzed the rankings produced by the Linear System Formulation. The Results and Analysis of the Linear System Formulation is given in the next section.

10.2 Results of Linear System Formulation

The rankings produced by the Linear System Formulation was almost similar to the rankings produced by the power method. The top 20 rankings from the Linear System Formulation are given in Table 10.1.

Names	Ranking Score
Muthurani MI Ashwin	0.076628086
Nithya Aravind	0.039699018
Rajesh Krishnan	0.039595379
Prakash Palanisamy	0.036468235
Ragavenderan Venkatesan	0.036405677
Janeni Vaidyanathan	0.036336098
Nanda Kumar	0.034201509
Bharathi Priyaa Thangamani	0.032093638
Mageshwaran Mohan	0.031731191
Mahesh Kumar	0.030832044
Mano Bharathi	0.030799899
Sowmi Paramasivam	0.030301659
Krishna Kumar	0.028547412
Iyswarya Abayamani	0.027842657
Prem Prakash	0.027546384
Yesudass Veeraiyan	0.027283654
Ramya S Krishnan	0.027221665
Nitesh Dakaliya	0.027184098
SeethaLakshmi Kuthalingam	0.02697951
Agni Varma	0.026714951

Table 10.1: Top 20 Friends and their Ranking Score

The results obtained from the Linear System Formulation highlight the important connections among the many connection present in the Facebook Friends Network. It is also very interesting to see that the results obtained from this project is highly accurate. The top 20 connections obtained are indeed the the most important connections. When this ranking list was compared with the ranking produced by the power method, it was very interesting to see that the rankings produced were not exactly the same. The ranks of some of the nodes were interchanged. The comparison between the results obtained from both the methods are given in Table 10.2. The nodes for which the rankings have been interchanged have been highlighted in bold. These changes maybe due to the fact that the ranking scores have very small differences in them (in the order of 0.00001) and thus the Linear System Formulation ranked some of the nodes higher than some nodes.

Names	Power Method Ranking Score	Power Rank	Linear Rank
Muthurani MI Ashwin	0.009711826	1	1
Nithya Aravind	**0.0056592**	**3**	**2**
Rajesh Krishnan	**0.005699151**	**2**	**3**
Prakash Palanisamy	0.005331969	4	4
Ragavenderan Venkatesan	**0.005232965**	**6**	**5**
Janeni Vaidyanathan	**0.005260252**	**5**	**6**
Nanda Kumar	0.0050242209	7	7
Bharathi Priyaa Thangamani	0.004645565	8	8
Mageshwaran Mohan	0.004555811	9	9
Mahesh Kumar	0.00450682	11	10
Mano Bharathi	0.004552256	10	11
Sowmi Paramasivam	0.004465793	12	12
Krishna Kumar	0.004218605	13	13
Iyswarya Abayamani	0.00406649	14	14
Prem Prakash	0.004038717	15	15
Yesudass Veeraiyan	0.004035948	16	16
Ramya S Krishnan	**0.003962702**	**18**	**17**
SeethaLakshmi Kuthalingam	**0.003985383**	**17**	**19**
Agni Varma	**0.003947713**	**19**	**20**

Table 10.2: Comparison between results obtained from Power Method and Linear Method

10.3 Tinkering with the Linear System Formulation

There are many modifications possible with the PageRank algorithm. Changing the value of α is one of the modifications possible. The main reason for using $\alpha = 0.85$ is for the fast convergence of the power method. As the Linear System Formulation is just forward substitution, changing the value of α does not affect the speed of computation of Linear System Formulation algorithm that much. Thus we can conclude that changing the value of α is not as significant in Linear System Formulation as it is in the Power Method in terms of speed. The rankings however will change.

When the value of α was changed, the ranking score for the nodes changed. This is pretty intuitive as we multiply α with the matrix, thereby changing its value.

Names	Linear Rank Score ($\alpha = 0.01$)	Linear Rank ($\alpha = 0.01$)	Linear Rank ($\alpha = 0.85$)
Muthurani MI Ashwin	0.002268976	1	1
Nithya Aravind	0.002081833	2	2
Ragavenderan Venkatesan	**0.002079405**	**3**	**5**
Rajesh Krishnan	**0.002078473**	**4**	**3**
Mageshwaran Mohan	**0.002068659**	**5**	**9**
Janeni Vaidyanathan	0.002068286	6	6
Prakash Palanisamy	**0.002064693**	**7**	**4**
Bharathi Priyaa Thangamani	0.002060216	8	8
Mahesh Kumar	**0.002057114**	**9**	**10**
Aishwarya Kothandaraman	**0.002054763**	**10**	**22**
Nanda Kumar	**0.002054377**	**11**	**7**
Sowmi Paramasivam	0.002049049	12	12
Ramya S Krishnan	**0.002048449**	**13**	**17**
Nagavenkata Apparao Pulluri	**0.002047469**	**14**	**93**
Nitesh Dakaliya	**0.002046702**	**15**	**18**
Mano Bharathi	**0.002046265**	**16**	**11**
Iyswarya Abayamani	**0.002045272**	**17**	**14**
Lloyd Anto	**0.002045065**	**18**	**76**
Guhan Balasubramanian	**0.002044835**	**19**	**21**
Amrutha Isakki	**0.002044305**	**20**	**278**

Table 10.3: Comparison between results obtained from $\alpha = 0.85$ and $\alpha = 0.01$

As we can see from the table, that a change in ranking as well as the change in the ranking scores. The new rank generated when the value of $\alpha = 0.01$ is not at all accurate and deviates from the original ranking by a large factor.

The nodes for which the ranking changed have been highlighted in bold. This table indicates the importance of choosing α as 0.85 or close to 0.85.

11 Conclusion for PageRank - Linear System Method

It is very evident from the above results and observations that the Linear System Formulation is an excellent method when we do not want to step into the Markov realm. The method converts the problem into a linear system and exploits the existence of dangling nodes to compute the stationary vector. We also looked at a couple of cases where the value of α is changed and how it changes the ranking of the nodes. We see that $\alpha = 0.85$ is a good value compared to $\alpha = 0.01$ with respect to the Linear System Formulation. Reducing or increasing the value of α does not help speed up the computation for this method and only changes the rankings. This is because Linear System Formulation is just a forward substitution and does not have any convergence. According to the author, the ranking produced when the value of $\alpha = 0.85$ was more accurate than the when $\alpha = 0.01$. The author reported that the system performs best when $\alpha = 0.99$, that is when α is at its highest possible value.

This project helped us understand the working of the PageRank algorithm in great detail. We understood the various changes possible to the algorithm. We were able to apply the algorithm to our own Facebook data and were able to infer various results from it.