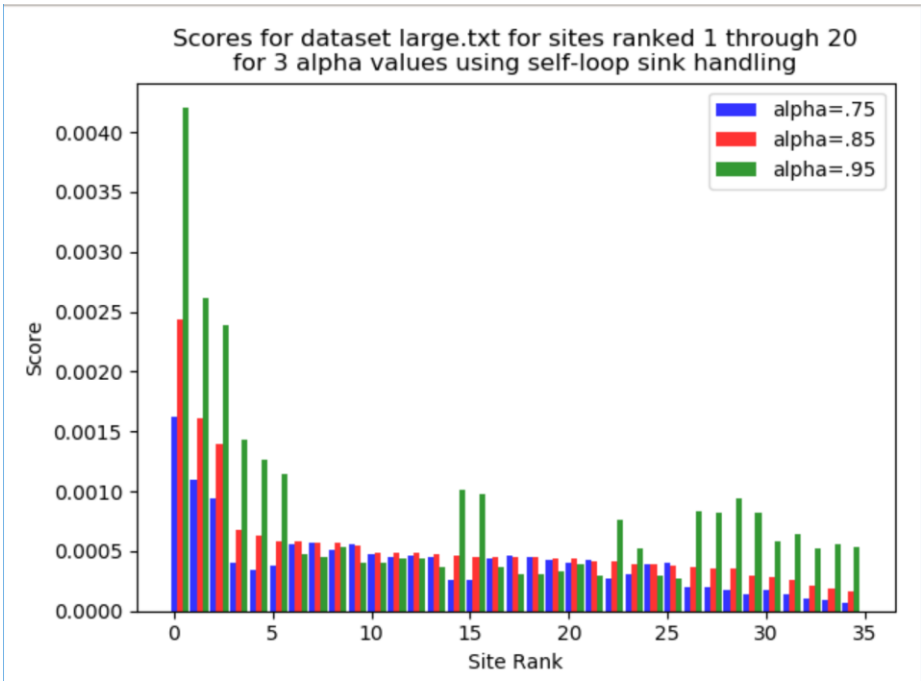
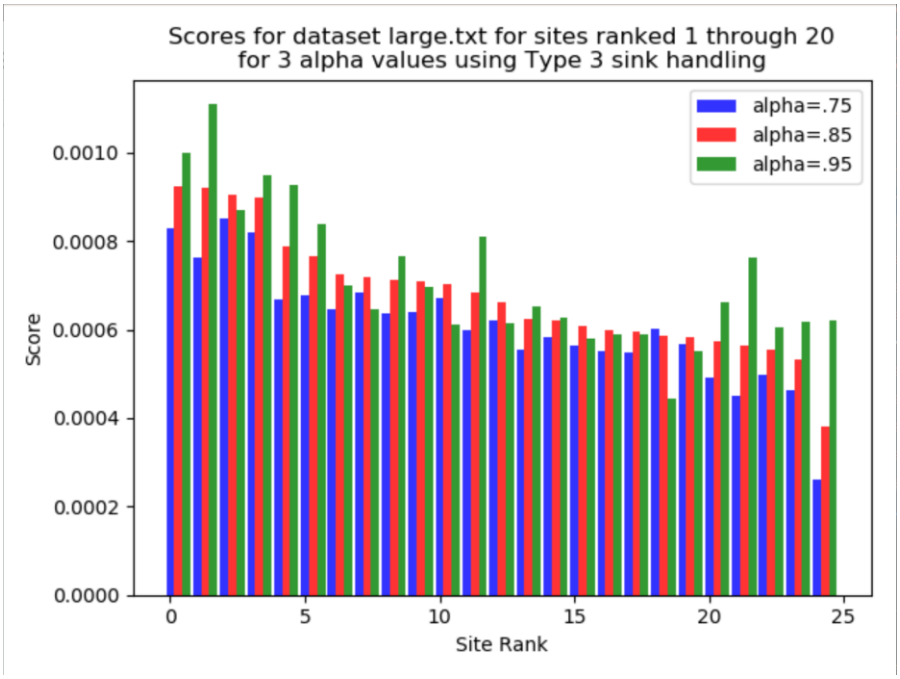
PageRank Project Report

Name : Joshua Ramer

Graph of results for 3 alphas for self-loop sink handling



Graph of results for 3 alphas for Type 3 sink handling



PageRank Project Report

Name : Joshua Ramer

Plots Discussion :

I. What do the plots show? What do the values represent?
   a. The plots show the probability at any given time during a random web browsing session of being on any of the top ranked pages; hence the name page rank algorithm. Each value corresponds to a particular node's rank in the graph calculated with different alpha values.

II. What does increasing alpha toward 1.0 mean? Why do the results change with changing alpha?
   a. As alpha approaches one, the probability contributed by visiting another page at random approaches zero. Simultaneously, increasing significance is assigned to the random walk on adjacent nodes. This results in higher concentrations of probability in parts of the graph that do not have outgoing links. These parts are strongly connected components and in aggregate, syncs. Why? We have decreased the probability of selecting a page at random and these components have no out links. That is why we see so much of the probability in the 'green' run above collected in the top page ranks.

III. Why does changing alpha affect runtime?
   a. As alpha increases, more probability is assigned to each page from its adjacent 'inList' connections. There are potentially many nodes in each 'inList' which results in large changes in probability which cause page rank to take longer to converge. Another way to look at it is that more probability must accumulate into these larger values from the entire graph. That will take more random walks and random page visits than the lower alpha values, 'blue' and 'red', which more evenly distribute probability throughout the graph.

IV. How does the sink handling strategy impact the results and the runtime?
   a. The self-loop sync strategy ensures that we don't lose probability through sync nodes when they do not visit a webpage at random by allowing them to visit themselves. We must also give all the other nodes this same ability to avoid causing probability to over accumulate in sync nodes. This strategy exacerbates the overly large alpha value problem and the result is on display in the top graph. The largest over accumulation of probabilities with respect to the rest of the graph occur in this scenario.
   b. Type-3 sync handling ensures that every node receives an equal share of the probability accumulated in sync nodes during each iteration. This results in the probability being more evenly distributed throughout the graph whilst still achieving a ranking. We also observe that the problem of probability of over accumulating in sync SCC's is dramatically reduced.
   c. The runtime of the self-looping sync strategy is faster in part due to fact that we don't need to loop through sync nodes and calculate the overall sync node contribution in every iteration. Another consideration is that type-3 sync handling will need more iterations to have enough probability flow through the sync nodes and then be redistributed throughout the graph to enable the very balanced distribution it achieves.