PageRank Algorithm

Description

PageRank is an algorithm that rates webpages on the basis of the number of hyperlinks that lead to it. The process involves simulating a random walk over all connected webpages which mathematically reduces to finding the principal left eigenvector of the probability matrix. The eigenvector can be found using numpy's linear algebra package or using the power iteration method (this does not always converge and the convergence checking condition has also been implemented).

Running the assignment

1. Open the .ipynb file in Jupyter notebook.
2. Run all the cells in the order given in the notebook.
3. In the same folder as the .ipynb file, create a .txt file with the filename of your choice.
4. In the .txt file, provide input in the following format: Line 1: A single integer (n) denoting the number of number of nodes Line 2: A single integer (e) denoting the number of number of edges Following e lines: Two integers (a, b) denoting the existence of an edge from node a to node b
5. Add a new code cell at the end of the .ipynb file and pass the filename of the .txt file as a parameter, along with whether teleportation must be implemented and the value of alpha if it is.

Runtime Analysis

For small values of n, both methods are extremely fast due to numpy's optimisation of matrix multiplication. For n=4, the maximum number of links is 16 (all nodes connected to themselves and all other nodes). On testing the algorithm with varying inputs from e=6 to e=16, the average runtime was almost constant at 0.1s for all inputs. While there were a couple of spikes of 0.8s and 0.6s for some inputs, these were not repeated and we believe the cause is most likely the processor and not the code.

HITS Algorithm

Description:

Hyperlink Induced Topic Search (HITS) Algorithm is a Link Analysis Algorithm that rates webpages. This algorithm is used to the web link-structures to discover and rank the webpages relevant for a particular search. HITS uses hubs and authorities to define a recursive relationship between webpages. This function calculates the hubs and authority score for each node in the graph.

Runtime Analysis

The Time complexity of HITS Algorithm implemented by us is O(number of iterations * number of edges in the graph).

How to run this assignment

1. Open the .ipynb file in Jupyter notebook.
2. Run all the cells in the order given in the notebook.
3. You will be asked to enter your query. Enter your query. The Query can be of multiple words.
4. You should run driver_AND function if you want your sub graph to be made using AND operations on tokenized query, if you want your sub graph to be made using OR Operations on tokenized query, use driver_OR.
5. Run the subsequent cells.
6. A Visual plot of sub graph will be plotted and details such as Number of nodes and edges in the subgraph will be printed and Adjecency list will also be printed.
7. One of the blocks will be calling HITS function which will return the hub and authority score.
8. Subsequent blocks will show the nodes in descending order of their converged hub and Authority values.
9. Subsequent block will show and plot the hub and authority error history and show that they have converged.

Made By -

1. Abhisht Rustagi (2020A7PS1891H)
2. Penugonda Satya Sohan (2020A7PS0190H)
3. Rohith Paul (2020A7PS2044H)