Implementation of Centrality Index
University of Delaware
Fall 2023
Emmanuel Adebayo & Gabriel Cofre

Homework 3

In this homework, the Page Rank algorithm will be implemented

1 Documentation

The Page Rank Algorithm, developed by Larry Page and Sergey Brin at Google, is a fundamental algorithm used in web search engines to rank web pages based on their importance and relevance.

1.1 Algorithm Description

The PageRank algorithm uses the following principles

- Each page is treated as a node in a graph
- Links between pages are represented as edges in the graph
- Page Rank is calculated iteratively, with each page's rank influenced by ranks of pages linking to it.

To calculate the Page Rank, we first calculate the Stochastic matrix from the Adjacency matrix. We then calculate the Transition matrix Q which is given by

$$Q = (1-\alpha)\,A + \frac{\alpha}{N}1_N \tag{1}$$

Where:

α is the damping factor. Typically 0.85 (https://doi.org/10.1016/j.jda.2009.11.001)
N is the total number of Nodes
1_N is a $n \times n$ matrix such that $a_{ij} = 1$, $\forall 1 \leq i, j \leq N$
The Equation of Pagerank is then Implemented using the Power method:

$$\lim_{k\to\infty} Q^k u = p \tag{2}$$

Steady-state Distribution

$$Qp = p \tag{3}$$

where p is the page rank, u is initialized vector

1.2 Implementation

Our Page Rank implementation is written in Python which consist of two main functions:

- *adjency_matrix(nodes):* We create a matrix in function with an arbitrary number of nodes to test the performance of the PageRank algorithm, our output is a list of weighted edges (generally 1s).
- $pagerank(adjacency\_matrix, \alpha, max\_iterations, tolerance)$: We use the adjency matrix in a list format, α is the damping factor, an arbitrary number of maximum of iterations and a tolerance threshold where it converges.

The PageRank algorithm is iteratively updating based on 1

1.3 Usage Instruction

To Implement the PageRank, follow these steps;

- Intialze the file path: for example *C:/Users/DELL/Desktop/All Python Codes/qpband.mtx*
- Prepare an adjacency matrix that represent the link structure
- Implement the PageRank Algorithm

2 2 Examples of sparse networks size 10-20k nodes

2.1 Data used

In this section, we used data from *sparse.tamu.edu*

- Data1: GHSindef/qpband QPBAND augmented system (20,000 nodes)
- Data2: HB/bcsstk29 S STIFFNESS MATRIX FOR BUCKLING MODEL OF THE 767 REAR BULKHEAD - PATTER (13,992 nodes)

2.2 Results

Aftert running the PageRank algorithm using our implementation, we obtain the PageRank scores for each page in the network. This scores reflects the importance of each page within the network.

2.2.1 PageRank Score Results

Since the PageRank scores are numerical values associated with each page in the network. Higher PageRank therefore indicates greater importance or relevance within the network.

For Data1:
The number of Iterations is 4. The time it took for the Iterations to converge is 0:00:01.001039
Highest: Node 1: 5.650828363191949e-05
Lowest: Node 13433: 3.359199964635407e-05

For Date2:
The number of Iterations is 4. The time it took for the Iterations to converge is 0:00:01.338438
Highest: Node 2445: 7.978928226311771e-05 Lowest: Node 12445: 6.622610956052168e-05

3 References

- Network Graph and Page Rank Algorithm
- The Power Method