Documentation for Q2)

Problem Statement:

In the given problem statement, we have to implement the PageRank and authority-hub algorithm to calculate the most relevant pages in the given dataset. For this, we have used the Wikivote data consisting of 7115 nodes and 103689 edges.

Import and Graph Creation:

First, we use standard file reading in Python to import the data. Then, we split the data into two parts: the Information part and the Graph part. The Information part contains metadata and statistics about the data, which we store in a variable called meta. The Graph part contains the connection information, which we send to a pandas dataframe. This dataframe has two columns: "ToNode" and "FromNode". Next, we use the networkx library in Python to create a graph from the pandas dataframe. To achieve this, we call the nx.from_pandas_edgelist function. Finally, we convert the resulting graph to a directed graph.

PageRank

The following paragraph is based on information from Wikipedia. At the start of the page rank algorithm, all nodes are assigned the same probability of being the most relevant page. To calculate the rank of each node, we consider the number of outgoing edges from that node, and divide its current probability by the total number of outgoing edges. We then add this value to the probabilities of all nodes that currently have a score of 0 and some fixed probability. This process is repeated for all nodes in the graph.

Summarizing the formula:
$PR(a) = \Sigma PR(i)/L(i)$ where L(i) is the total number of outgoing edges for that node and PR(i) is the current pagerank score.

In addition, we calculate the Mean Squared Error (MSE) between the newly calculated PageRank scores and the old scores. This is used to halt the convergence of the algorithm when the error becomes smaller than 1e-15. By default, the algorithm runs for 100 iterations, but it typically converges after 20-25 iterations.

Authority and Hub:

Based on information from Wikipedia, the Authority score of a page is based on the number of incoming edges it has from other pages, while the Hub score is based on the number of outgoing edges it has to other pages. Initially, both the Hub and Authority scores for all pages are set to one. To keep track of the previous scores, a buffer for both the Authority and Hub scores is maintained.

Next, we compute the Mean Squared Error (MSE) between the current values for both the Authority and Hub scores, and update the buffer accordingly. The algorithm continues to run until the MSE for both scores is greater than 1e-20, at which point it terminates. By default, the algorithm has a maximum of 100 iterations.

Results:

top 10 rank scores :

2565	0.0043372949187308815
11	0.003017206269367328
766	0.002968177479349323
457	0.002963411320667381
4037	0.002878218886740526
1549	0.0028581648714845506
1166	0.002669208905008099
2688	0.0023843472728713416
15	0.002163159726354969
1374	0.002131987766043142

top 10 authority scores :

2565	0.15769611748358103
766	0.13015243025685455
1549	0.12938941353080033
1166	0.11950594168986171
2688	0.11008403659853248
457	0.10999186611635883
3352	0.09179709631226124
11	0.08956574261869124
1151	0.08717924518500951
1374	0.08692950770481205

top 10 hub scores :

2565	0.157696117537377
766	0.13015243029945367
1549	0.12938941344572305
1166	0.11950594165584667
2688	0.11008403661789759
457	0.10999186615700852
3352	0.09179709627666102
11	0.08956574247014454
1151	0.08717924513642718
1374	0.08692950771109112