CSE508 Information Retrieval

Winter 2023

Assignment-1

Submitted by:
Sehban Fazili (MT21143)
Asmita Mukherjee (MT21115)
Amaan Khan (2020388)

Question 1 - Link Analysis

Represent the network in terms of its ‘adjacency matrix’ as well as ‘edge list’

Edge list:

Methodology:

Make a dictionary consisting of keys of all nodes.
Each node is associated with a list
Read each line of the file and split each line by ‘\t’
’ Append the node2 to the list associated list of node 1

Result:

```
'5346': ['1658',
 '4822',
 '5052',
 '6864',
 '7689',
 '7926',
 '9124',
 '10268',
 '12971',
 '15159',
 '18600',
 '20421',
 '20886',
 '21048',
 '22393',
 '23186',
 '23214',
 '23298',
 '23945',
 '24939'],
'15159': ['5052', '5346', '20421', '22393'],
```

Adjacency matrix:

Methodology:

1) Make a dataframe where all the nodes make up the rows and columns. Dimension: Nos_of_nodes * Nos_of_nodes
2) For each edge in the edge list indicate 1 when there is an edge between df[node1][node2]

Result:

	3466	10310	5052	5346	15159	19640	10243	18648	16470	17822	...	16590	16730	19957	21848	22320	5774	4836	17464	10154	11113
3466	0	1	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
10310	1	0	0	0	0	1	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
5052	0	0	0	1	1	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
5346	0	0	1	0	1	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
15159	0	0	1	1	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...	...
5774	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
4836	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
17464	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
10154	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0
11113	0	0	0	0	0	0	0	0	0	0	...	0	0	0	0	0	0	0	0	0	0

5242 rows × 5242 columns

Number of Nodes

1. Number of Nodes

```
In [12]: 1 print("Nos of nodes in the graph",len(edge_list))
```

```
Nos of nodes in the graph 5242
```

Number of edges:

2. Number of Edges

```
print("Number of edges in the graph")
sum(len(val) for val in edge_list.values()) #Since it is a directed graph, there as many number of edges in edge list as in
```

```
Number of edges in the graph

28980
```

Avg In-degree

3. Avg In-degree

In [14]:
```
nos_of_nodes = len(edge_list)
```

In [19]:
```
sum_of_inc_degree = 0

#take sum of column, since df[row][col] indicates edge from row->col
for node in adj_matrix.columns:
    sum_of_inc_degree += sum(adj_matrix[node])

avg_in_deg = sum_of_inc_degree/nos_of_nodes
```

In [20]:
```
avg_in_deg
```

Out[20]:
```
5.528424265547501
```

Avg. Out-Degree

In [21]:
```
sum_of_out_degree = 0

#take sum of row , since df[row][col] indicates edge from row->col
for node in adj_matrix.index:
    sum_of_out_degree += sum(adj_matrix.loc[node])

avg_out_deg = sum_of_out_degree/nos_of_nodes
```

In [22]:
```
avg_out_deg
```

Out[22]:
```
5.528424265547501
```

As the graph is undirected hence it has the same avg indegree and avg outdegree

Node with Max In-degree

In [23]:
```
max_in_degree = 0
res_node = -1

for node in adj_matrix.columns:
    in_degree_of_node = sum(adj_matrix[node])

    if max_in_degree<in_degree_of_node:
        max_in_degree = in_degree_of_node
        res_node = node


print(f"The node {res_node} has the max_in_degree of {max_in_degree}")
```

```
The node 21012 has the max_in_degree of 81
```

Node with Max out-degree

In [25]:
```
max_out_degree = 0
res_node = -1

#df[row][col] indicates edge from row->col
for node in adj_matrix.index:
    out_degree_of_node = sum(adj_matrix.loc[node])
    if out_degree_of_node>max_out_degree:
        max_out_degree = out_degree_of_node
        res_node = node


print(f"The node {res_node} has the max out degree of {max_out_degree}")

```

```
The node 21012 has the max out degree of 81
```

The density of the network

In [29]:
```

nos_of_edges = sum(adj_matrix.apply(pd.value_counts).loc[1].values)
```

In [31]:
```
nos_of_edges = nos_of_edges/2 #since undirected
```

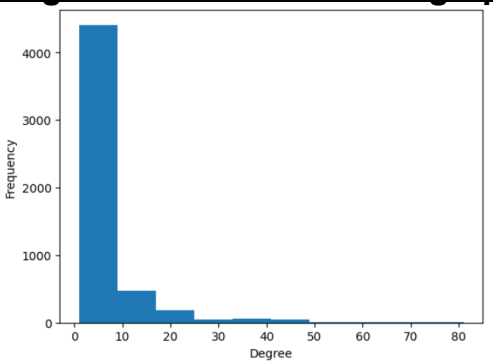
In [32]:
```
potential_edges = (nos_of_nodes*(nos_of_nodes-1))/2
```

In [33]:
```
density = nos_of_edges/potential_edges
```

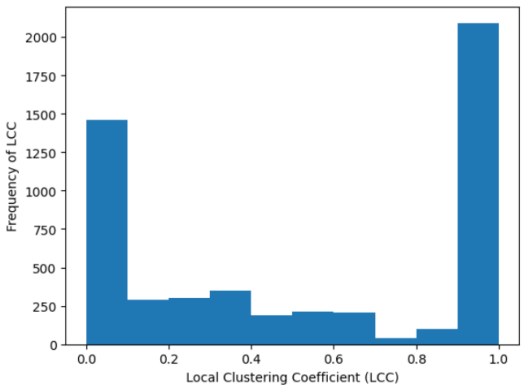
In [34]:
```
print(f"The density of the network is {density}")
```

```
The density of the network is 0.0010548414931401452
```

Degree distribution of the graph

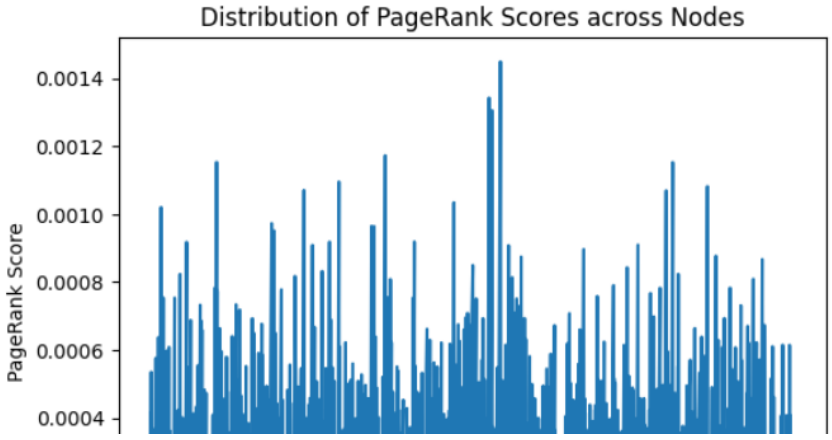


Clustering-coefficient distribution (lcc vs frequency of lcc) of the network.



Question 2 - PageRank, Hubs and Authority

PageRank score for each node



Authority and Hub score for each node