Information Retrieval

Assignment 3

Team 105

Course: CSE-508
Course Name: Information Retrieval
Instructor: Dr. Rajiv Ratn Shah

Deep Sharma - 2020370
Jaskaran Singh - 2020306
Pratyush Kumar - 2020454

Dataset Used **2**

1. Link Analysis **2**
Adjacency Matrix 2
Edge List 2
1.1. Number of nodes 2
1.2.Number of edges 2
1.3 Average in-degree of nodes 2
1.4 Average out-degree of nodes 3
1.5 Node with maximum in-degree 3
1.6 Node with maximum out-degree 3
1.7 Density of network 3
1.8 Degree distribution of the network 4
1.9 Clustering Coefficient distribution 4

2. PageRank, Hubs, and Authority **6**
2.1 PageRank score 6
2.2 Authority and Hub score 6
2.3 Comparison 6

Dataset Used

The dataset used for this assignment is facebook_combined, which is a directed network with 4039 nodes and 88234 edges.

1. Link Analysis

Adjacency Matrix

```
[[0 1 1 ... 0 0 0]
 [0 0 0 ... 0 0 0]
 [0 0 0 ... 0 0 0]
 ...
 [0 0 0 ... 0 0 0]
 [0 0 0 ... 0 0 0]
 [0 0 0 ... 0 0 0]]
```

Edge List

```
edges = list(zip(source_nodes, target_nodes))
print(edges)
```

1.1. Number of nodes

This was given in the dataset.

```
Number of nodes in graph: 6301
```

1.2.Number of edges

This was given in the dataset.

```
Number of edges in graph: 20777
```

1.3 Average in-degree of nodes

The average in-degree was calculated by summing the in-degrees for all the nodes and dividing it by the number of nodes.

```
in_degrees = [sum(adj_matrix[:, i]) for i in range(num_nodes)]
avg_in_degree = sum(in_degrees) / num_nodes
```

```
Avg In-degree: 3.2974131090303125
```

1.4 Average out-degree of nodes

The average in-degree was calculated by summing the out-degrees for all the nodes and dividing it by the number of nodes.

```
out_degrees = [sum(adj_matrix[i, :]) for i in range(num_nodes)]
avg_out_degree = sum(out_degrees) / num_nodes
```

```
Avg Out-degree: 3.2974131090303125
```

1.5 Node with maximum in-degree

We find the index of the node with the maximum in-degree.

```
max_in_degree_node = in_degrees.index(max(in_degrees))
```

```
Node with Max In-degree: 266
Degree of the node: 91
```

1.6 Node with maximum out-degree

We find the index of the node with the maximum out-degree.

```
max_out_degree_node = out_degrees.index(max(out_degrees))
```

```
Node with Max Out-degree: 5831
Degree of the node: 48
```

1.7 Density of network

The density of a directed graph is given by:

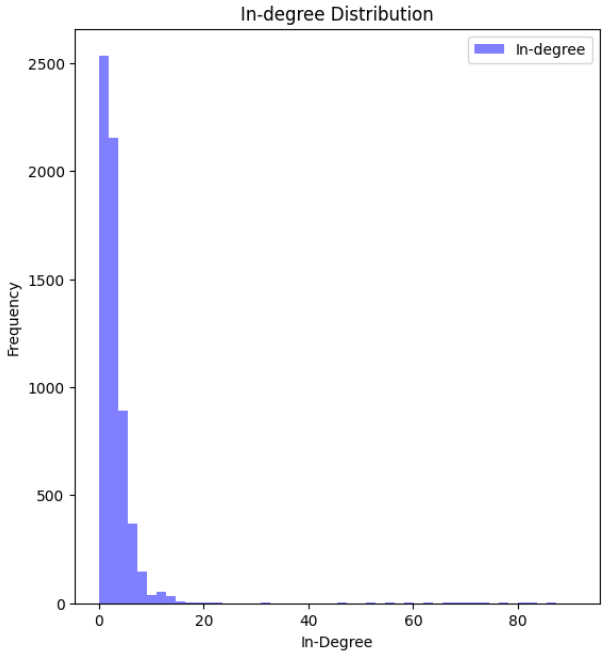
$$D = \frac{|E|}{|V|\left(|V| - 1\right)}$$

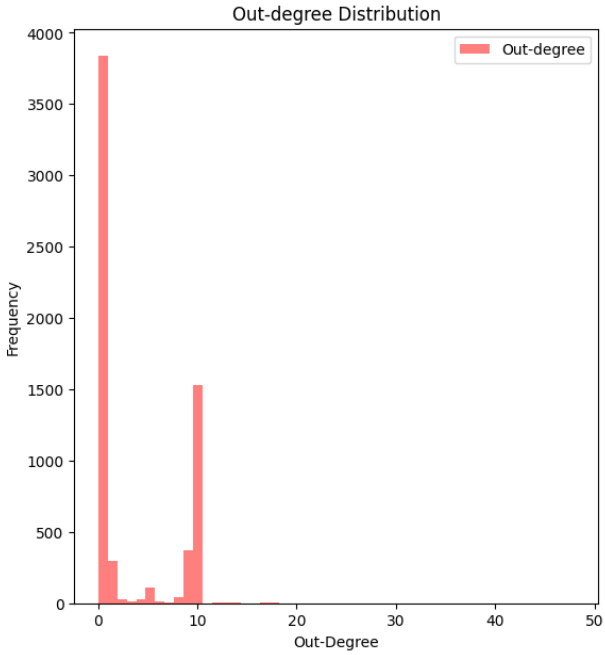
Here, D is the density of the directed graph, E is the number of edges, and V is the number of nodes.

```
density = num_edges / (num_nodes * (num_nodes - 1))
```

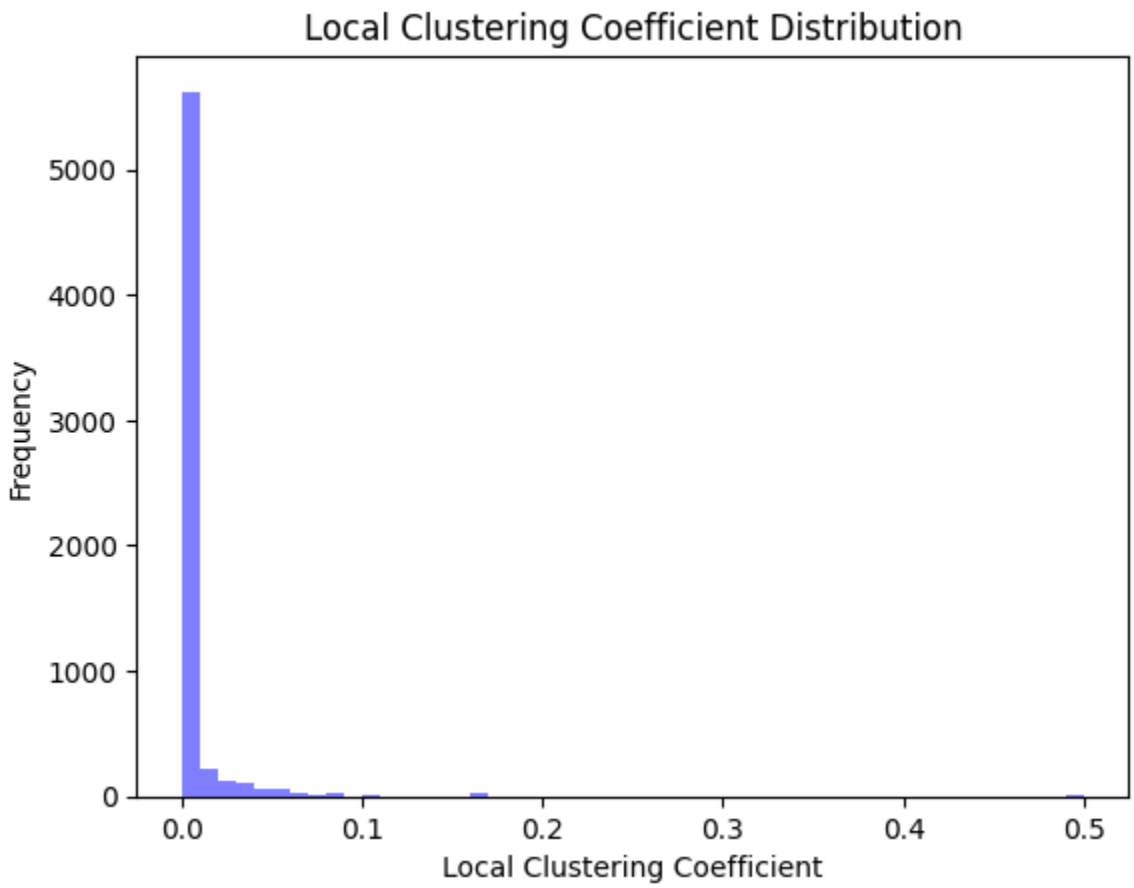
```
Density of the network: 0.0005233989061952878
```

1.8 Degree distribution of the network





1.9 Clustering Coefficient distribution



The formula for the clustering coefficient C_i for node i is given by:

$$C_i = \frac{|\{e_{jk} : v_j, v_k \in N_i, e_{jk} \in E\}|}{k_i(k_i - 1)}.$$

Here, E is the set of edges, and k_i is the number of nodes.

The neighborhood N_i for node v_i is defined as

$$N_i = \{v_j : e_{ij} \in E \vee e_{ji} \in E\}.$$

2. PageRank, Hubs, and Authority

2.1 PageRank score

The score is calculated using the pagerank function available in the Networkx library, which is based on the PageRank algorithm.

```
pagerank_scores = nx.pagerank(G)
for node, score in pagerank_scores.items():
    print(f'Node {node}: {score:.6f}')
```

2.2 Authority and Hub score

These are calculated using the hits function available in the Networkx library, which is based on the HITS algorithm.

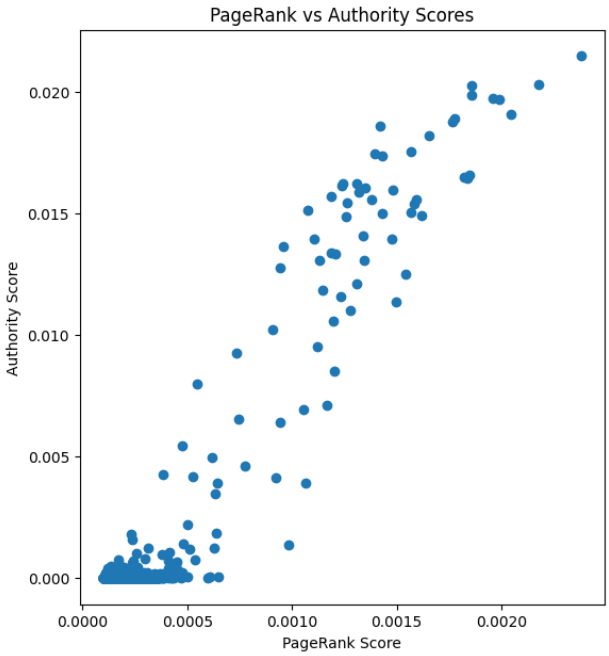
```
hubs, authorities = nx.hits(G)
for node, score in hubs.items():
    print(f'Node {node}: {score:.6f}')
for node, score in authorities.items():
    print(f'Node {node}: {score:.6f}')
```

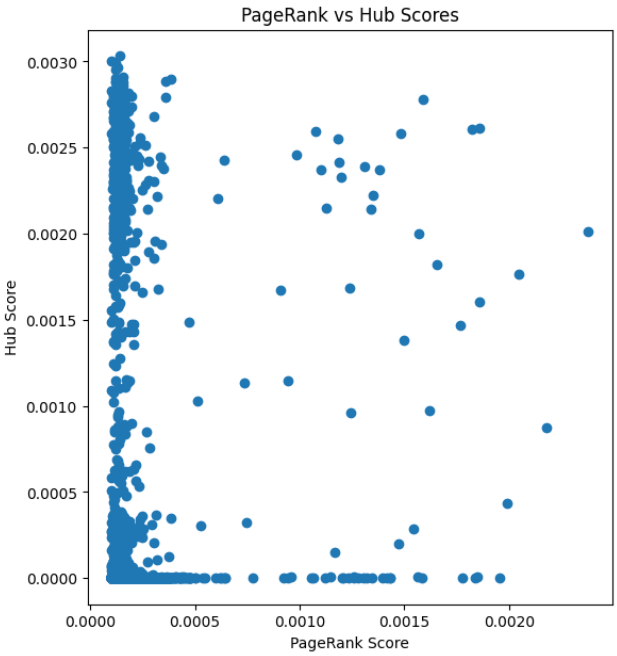
2.3 Comparison

The PageRank algorithm calculates a score for each node based on its importance in the network. The Authority score calculated by the HITS algorithm is based on the quality of its outgoing links, and its Hub score is based on the quality of its incoming links.

The PageRank scores represent the importance of each node in the network based on the concept that a node is important if other important nodes point to it. In contrast, the HITS algorithm calculates two scores for each node based on its role in the network: a node is considered an authority if other authoritative nodes point to it, and a node is considered a hub if it points to other hub nodes.

Comparing the results obtained from both algorithms can provide insight into the structure of the network and the roles that nodes play in it. For example, nodes with high PageRank scores are likely also to have high Authority scores in the HITS algorithm since they are important nodes that are pointed to by other important nodes. On the other hand, nodes with high Hub scores in the HITS algorithm are likely to have high out-degree in the PageRank algorithm since they are hubs that point to other important nodes.


PageRank vs Authority Scores
Authority Score
PageRank Score
0.000
0.005
0.010
0.015
0.020
0.0000
0.0005
0.0010
0.0015
0.0020


PageRank vs Hub Scores
Hub Score
PageRank Score
0.0000
0.0005
0.0010
0.0015
0.0020
0.0025
0.0030
0.0000
0.0005
0.0010
0.0015
0.0020