# INFORMATION RETRIEVAL(ASSIGNMENT-3)

GROUP MEMBERS:
Ankit Talreja Sahitya(MT22012)
Arohi Shrivastava(MT22017)
Ashutosh Choubey(MT22020)

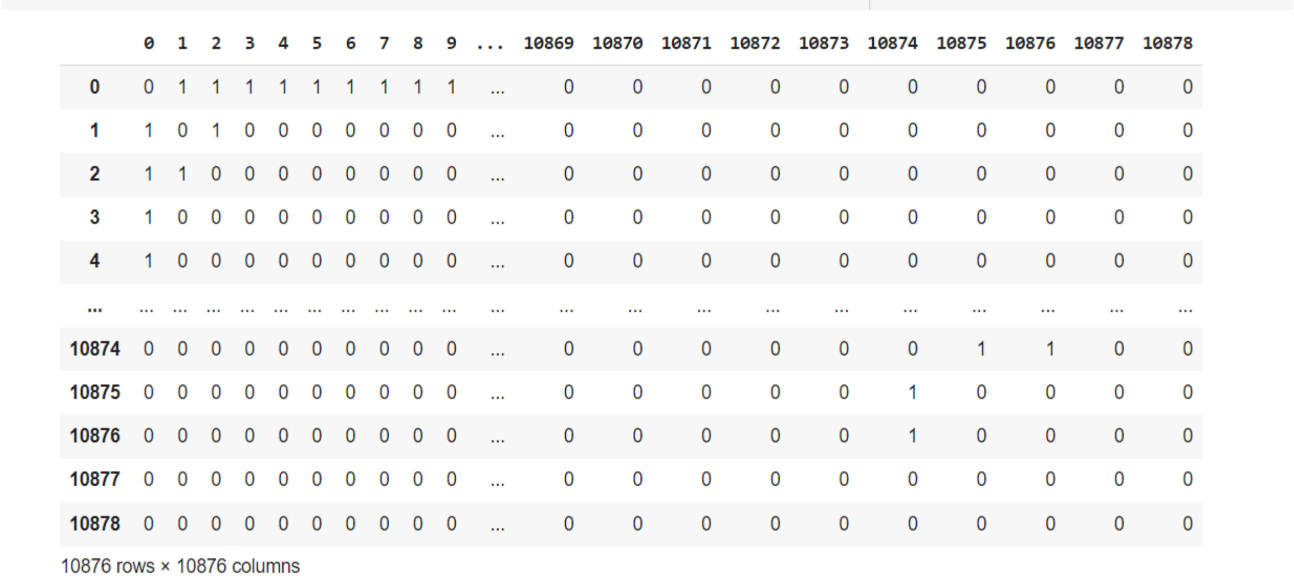
**LIBRARIES USED:**

- pandas : It provides tools for data cleaning, transformation, aggregation, and visualization, making it an essential tool for data scientists and analysts.
- Matplotlib : Matplotlib is a popular plotting library in Python.
- Prettytable : It is useful for displaying tabular data in a readable and organized format.
- tqdm : It is useful for tracking the progress of a long-running operation, such as a loop, and provides an easy way to visualize how much of the operation has been completed.
- networkx : python library for studying graph networks.

Answer-1:

Dataset chosen: Gnutella peer to peer network

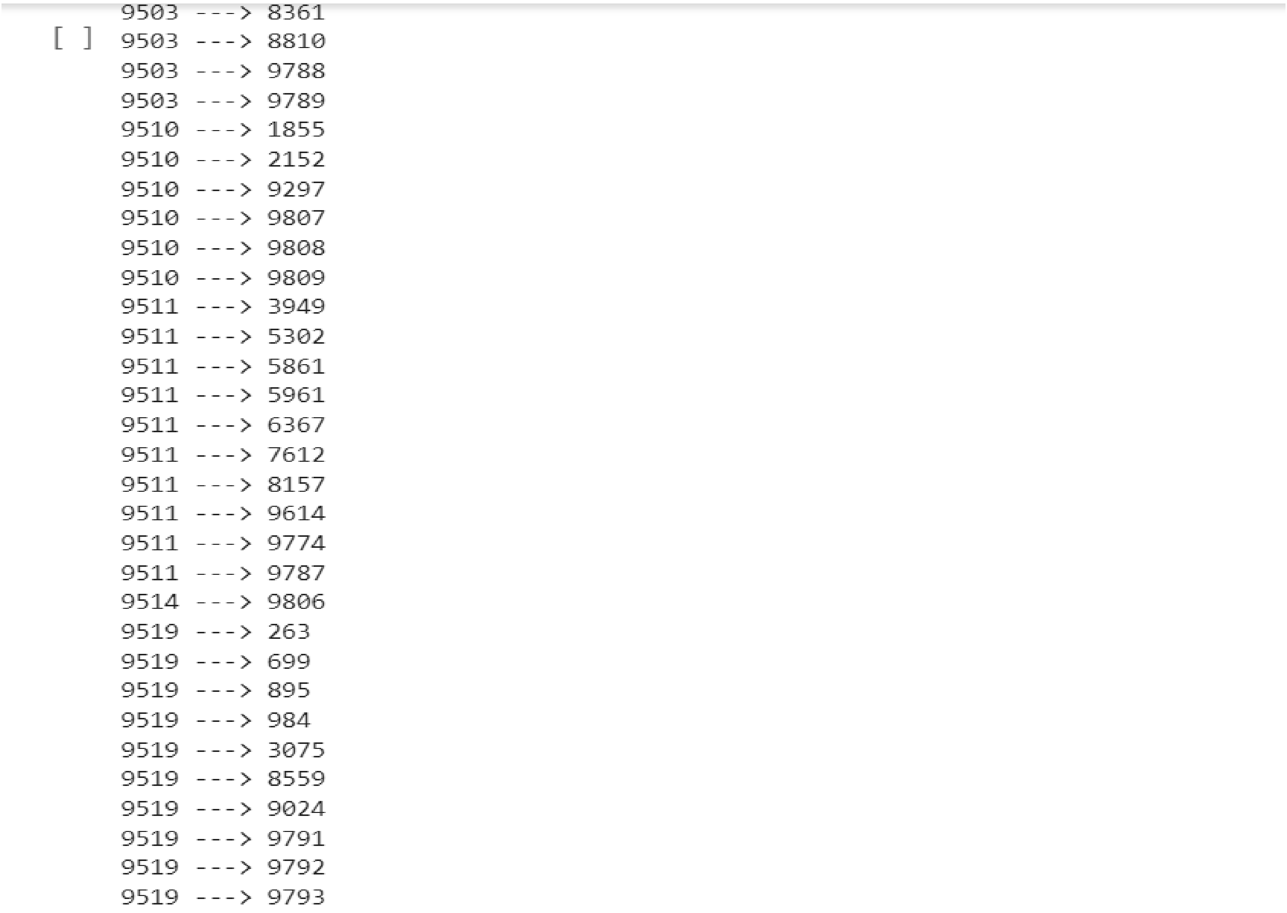
Network representation in the form of adjacency matrix. For an adjacency matrix, if there is an edge from node 1 to 2, then the adj_matrix [1][2]=1 else it will be 0. The value for adj_matrix [1][2] is 1 if node 1 links with node 2.



| | 0 | 1 | 2 | 3 | 4 | 5 | 6 | 7 | 8 | 9 | ... | 10869 | 10870 | 10871 | 10872 | 10873 | 10874 | 10875 | 10876 | 10877 | 10878 |
|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|---|
| **0** | 0 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | 1 | ... | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **1** | 1 | 0 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **2** | 1 | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **3** | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **4** | 1 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **...** | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... | ... |
| **10874** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 0 | 0 | 0 | 0 | 0 | 0 | 1 | 1 | 0 | 0 |
| **10875** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| **10876** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 0 | 0 | 0 | 0 | 0 | 1 | 0 | 0 | 0 | 0 |
| **10877** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |
| **10878** | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | ... | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 | 0 |

10876 rows × 10876 columns

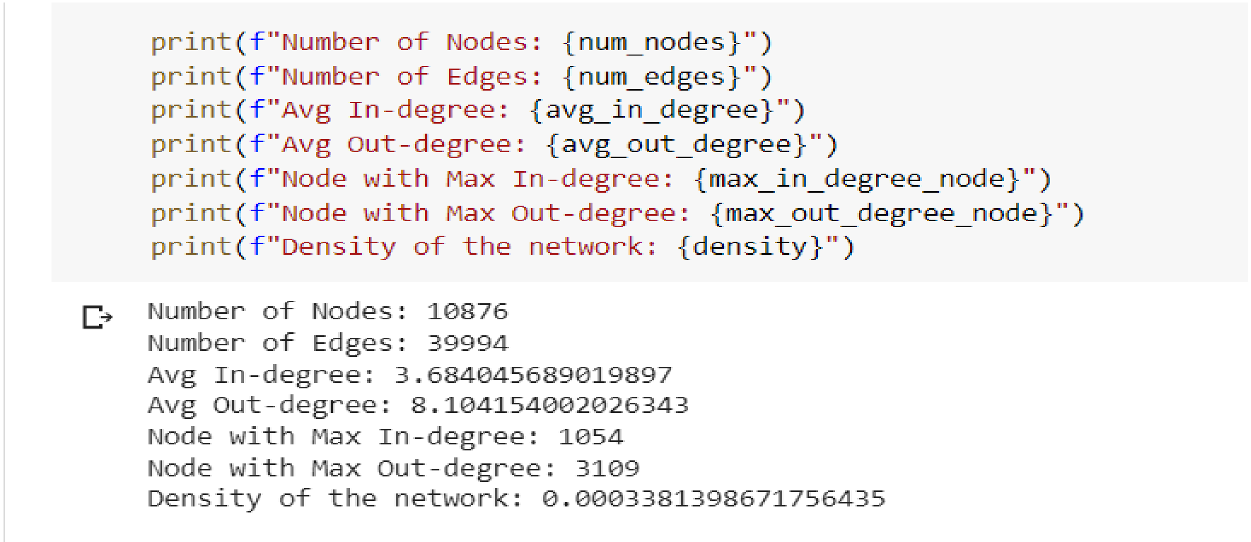
Edge representation of the peer to peer network:



```
9503 ---> 8361
9503 ---> 8810
9503 ---> 9788
9503 ---> 9789
9510 ---> 1855
9510 ---> 2152
9510 ---> 9297
9510 ---> 9807
9510 ---> 9808
9510 ---> 9809
9511 ---> 3949
9511 ---> 5302
9511 ---> 5861
9511 ---> 5961
9511 ---> 6367
9511 ---> 7612
9511 ---> 8157
9511 ---> 9614
9511 ---> 9774
9511 ---> 9787
9514 ---> 9806
9519 ---> 263
9519 ---> 699
9519 ---> 895
9519 ---> 984
9519 ---> 3075
9519 ---> 8559
9519 ---> 9024
9519 ---> 9791
9519 ---> 9792
9519 ---> 9793
```

Answer 1.1)

Dataset chosen: Gnutell peer to peer Network
Nodes in the network: no of people
Edges in the network: nodes who share between each other.



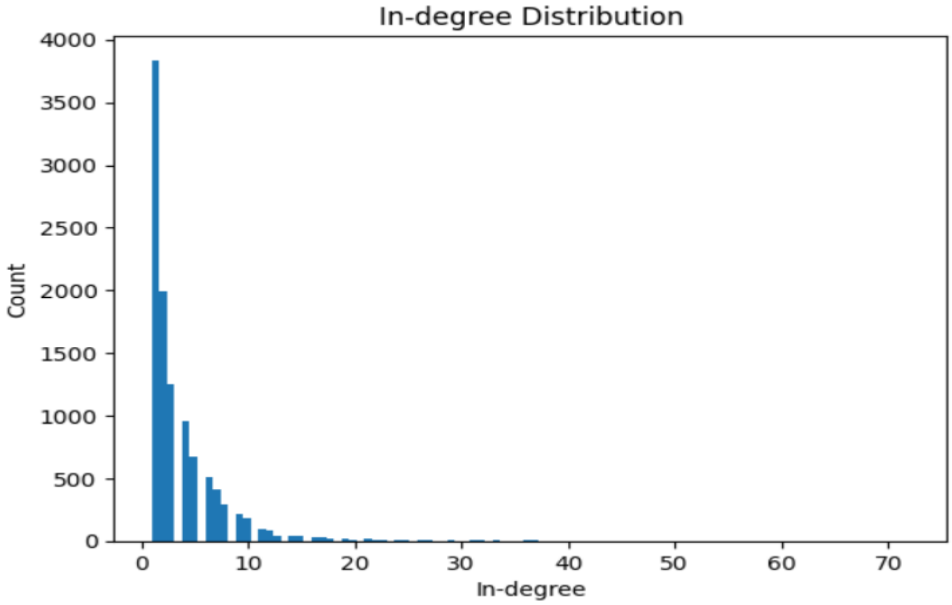
```
print(f"Number of Nodes: {num_nodes}")
print(f"Number of Edges: {num_edges}")
print(f"Avg In-degree: {avg_in_degree}")
print(f"Avg Out-degree: {avg_out_degree}")
print(f"Node with Max In-degree: {max_in_degree_node}")
print(f"Node with Max Out-degree: {max_out_degree_node}")
print(f"Density of the network: {density}")
```

```
Number of Nodes: 10876
Number of Edges: 39994
Avg In-degree: 3.684045689019897
Avg Out-degree: 8.104154002026343
Node with Max In-degree: 1054
Node with Max Out-degree: 3109
Density of the network: 0.0003381398671756435
```

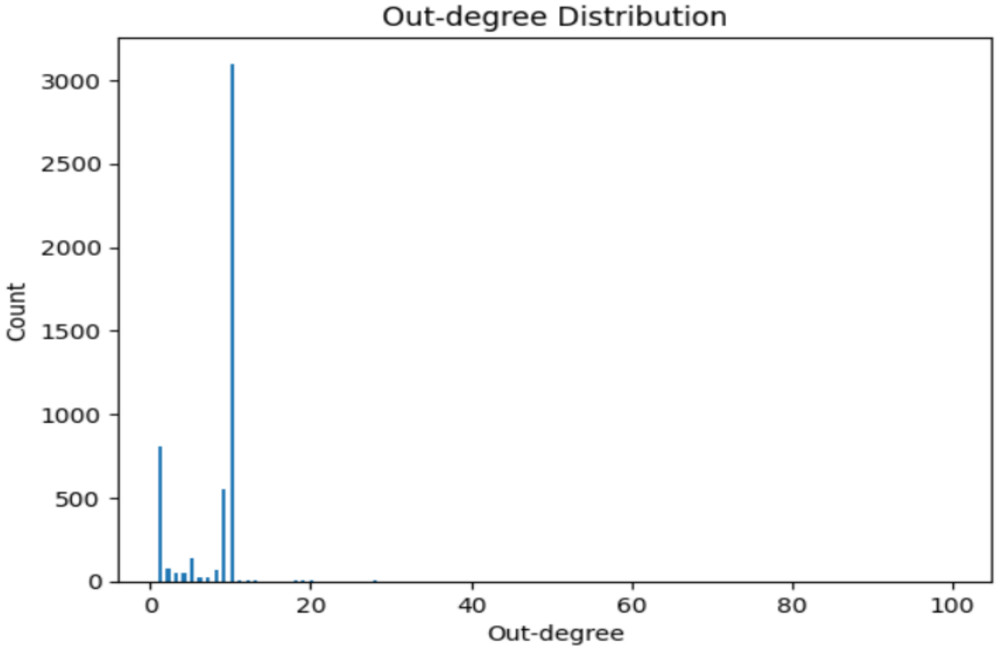
we have a directed graph, we can analyze its in-degree and out-degree distribution. The average in-degree and out-degree will be equal because nodes with a high in-degree will be balanced out by nodes with a high out-degree.

Network density is a measure of how connected a graph is. If the density is 0, then there are no edges in the network, while a density of 1 indicates a complete graph.

Tocalculate network density for a directed graph, we use the formula: total number ofedges divided by n times n-1, where n is the total number of nodes in the network.



Outdegree distribution :



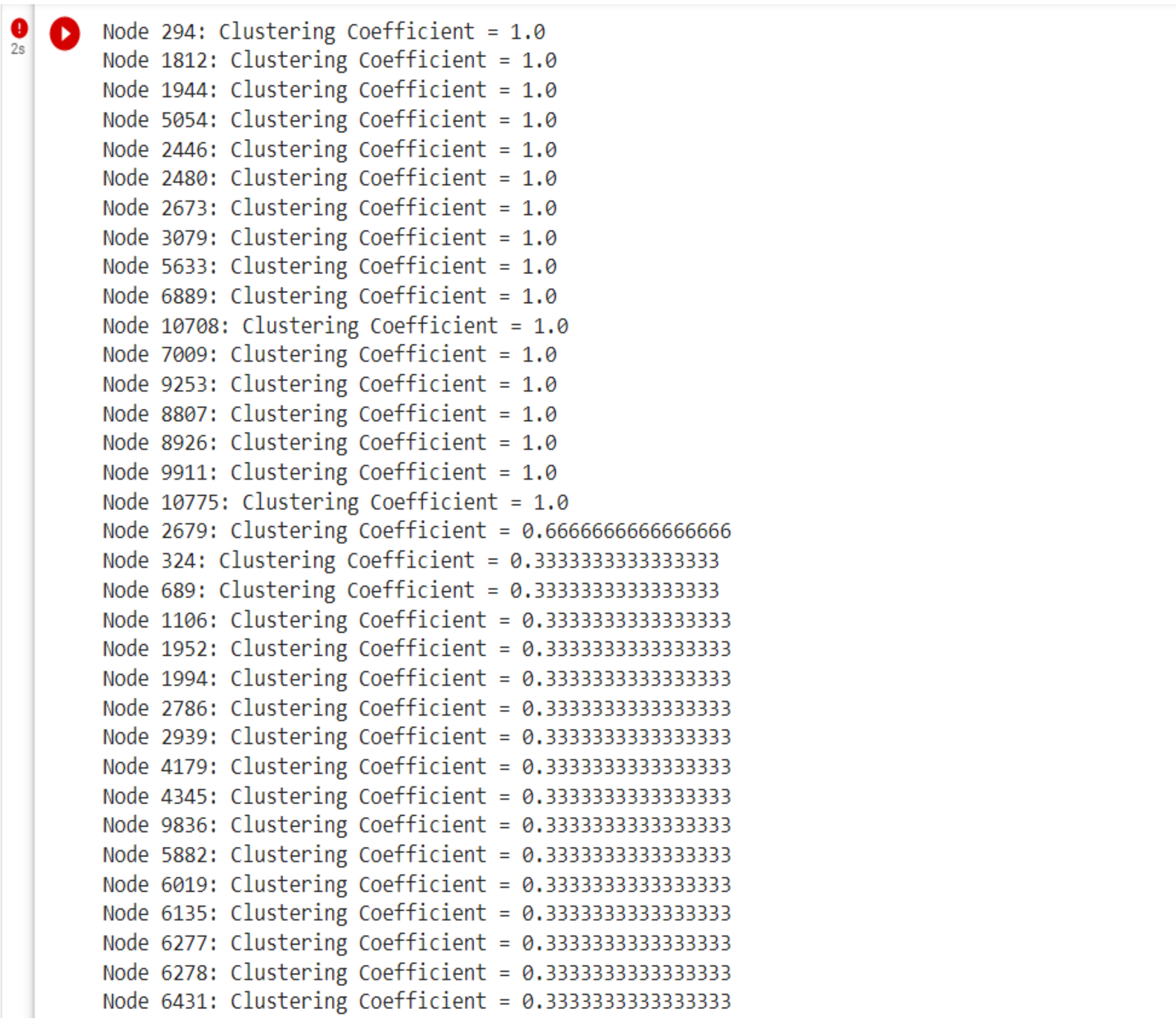
Answer 1.2)

We can calculate the clustering coefficient of each node in a graph. The clustering coefficient ranges between 0 and 1, with values closer to 1 indicating a higher level of certainty.

We can also count the number of nodes with a clustering coefficient of 0 and 1 in the graph.

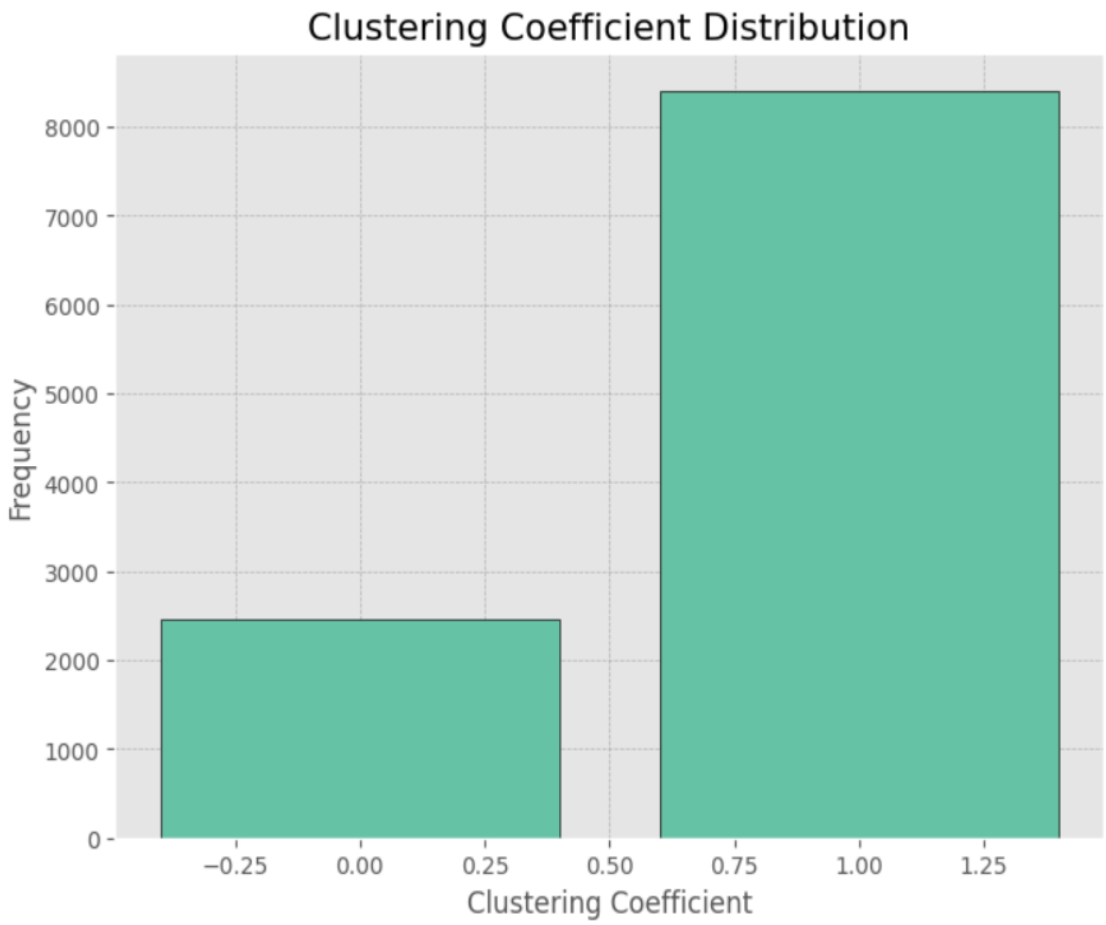
The overall clustering coefficient of the network can be determined by calculating the average clustering coefficient across all nodes.

To calculate the clustering coefficient of a directed graph, we use the formula: N divided by n times (n-1), where n is the total number of neighbors for the node, and N is the number of edges among those neighbors in the network.

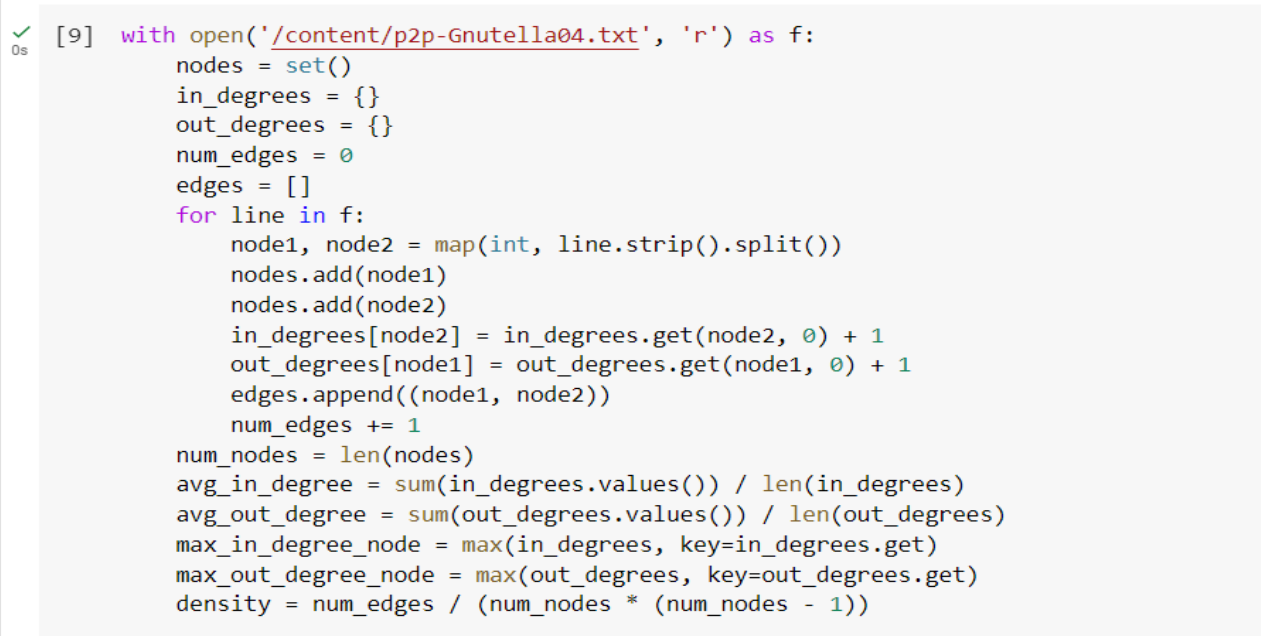


```
Node 294: Clustering Coefficient = 1.0
Node 1812: Clustering Coefficient = 1.0
Node 1944: Clustering Coefficient = 1.0
Node 5054: Clustering Coefficient = 1.0
Node 2446: Clustering Coefficient = 1.0
Node 2480: Clustering Coefficient = 1.0
Node 2673: Clustering Coefficient = 1.0
Node 3079: Clustering Coefficient = 1.0
Node 5633: Clustering Coefficient = 1.0
Node 6889: Clustering Coefficient = 1.0
Node 10708: Clustering Coefficient = 1.0
Node 7009: Clustering Coefficient = 1.0
Node 9253: Clustering Coefficient = 1.0
Node 8807: Clustering Coefficient = 1.0
Node 8926: Clustering Coefficient = 1.0
Node 9911: Clustering Coefficient = 1.0
Node 10775: Clustering Coefficient = 1.0
Node 2679: Clustering Coefficient = 0.6666666666666666
Node 324: Clustering Coefficient = 0.3333333333333333
Node 689: Clustering Coefficient = 0.3333333333333333
Node 1106: Clustering Coefficient = 0.3333333333333333
Node 1952: Clustering Coefficient = 0.3333333333333333
Node 1994: Clustering Coefficient = 0.3333333333333333
Node 2786: Clustering Coefficient = 0.3333333333333333
Node 2939: Clustering Coefficient = 0.3333333333333333
Node 4179: Clustering Coefficient = 0.3333333333333333
Node 4345: Clustering Coefficient = 0.3333333333333333
Node 9836: Clustering Coefficient = 0.3333333333333333
Node 5882: Clustering Coefficient = 0.3333333333333333
Node 6019: Clustering Coefficient = 0.3333333333333333
Node 6135: Clustering Coefficient = 0.3333333333333333
Node 6277: Clustering Coefficient = 0.3333333333333333
Node 6278: Clustering Coefficient = 0.3333333333333333
Node 6431: Clustering Coefficient = 0.3333333333333333
```

Distribution of Clustering Coefficients:



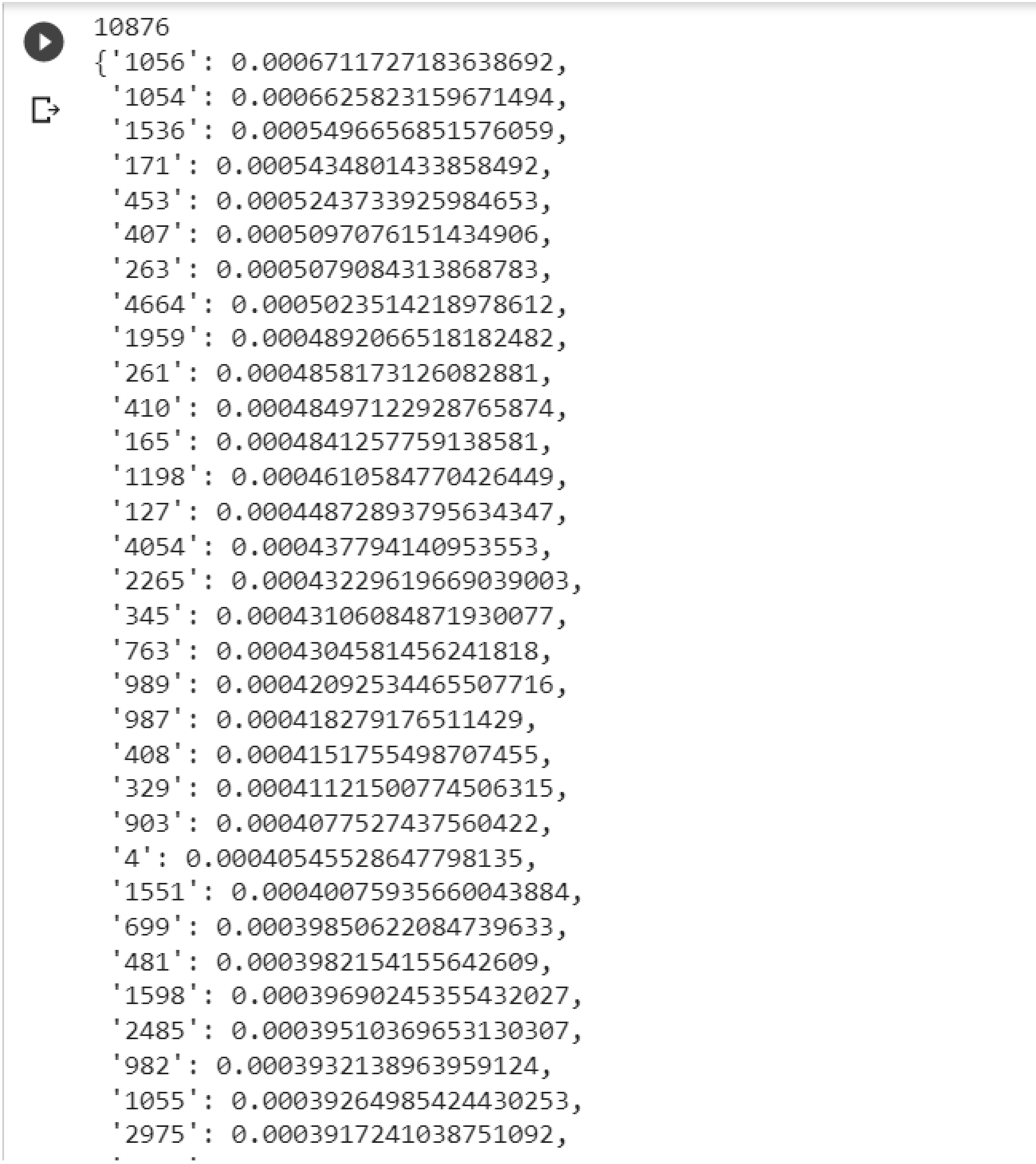
Formulas used in the question to calculate the metrics:



```
with open('/content/p2p-Gnutella04.txt', 'r') as f:
    nodes = set()
    in_degrees = {}
    out_degrees = {}
    num_edges = 0
    edges = []
    for line in f:
        node1, node2 = map(int, line.strip().split())
        nodes.add(node1)
        nodes.add(node2)
        in_degrees[node2] = in_degrees.get(node2, 0) + 1
        out_degrees[node1] = out_degrees.get(node1, 0) + 1
        edges.append((node1, node2))
        num_edges += 1
    num_nodes = len(nodes)
    avg_in_degree = sum(in_degrees.values()) / len(in_degrees)
    avg_out_degree = sum(out_degrees.values()) / len(out_degrees)
    max_in_degree_node = max(in_degrees, key=in_degrees.get)
    max_out_degree_node = max(out_degrees, key=out_degrees.get)
    density = num_edges / (num_nodes * (num_nodes - 1))
```

Answer-2)

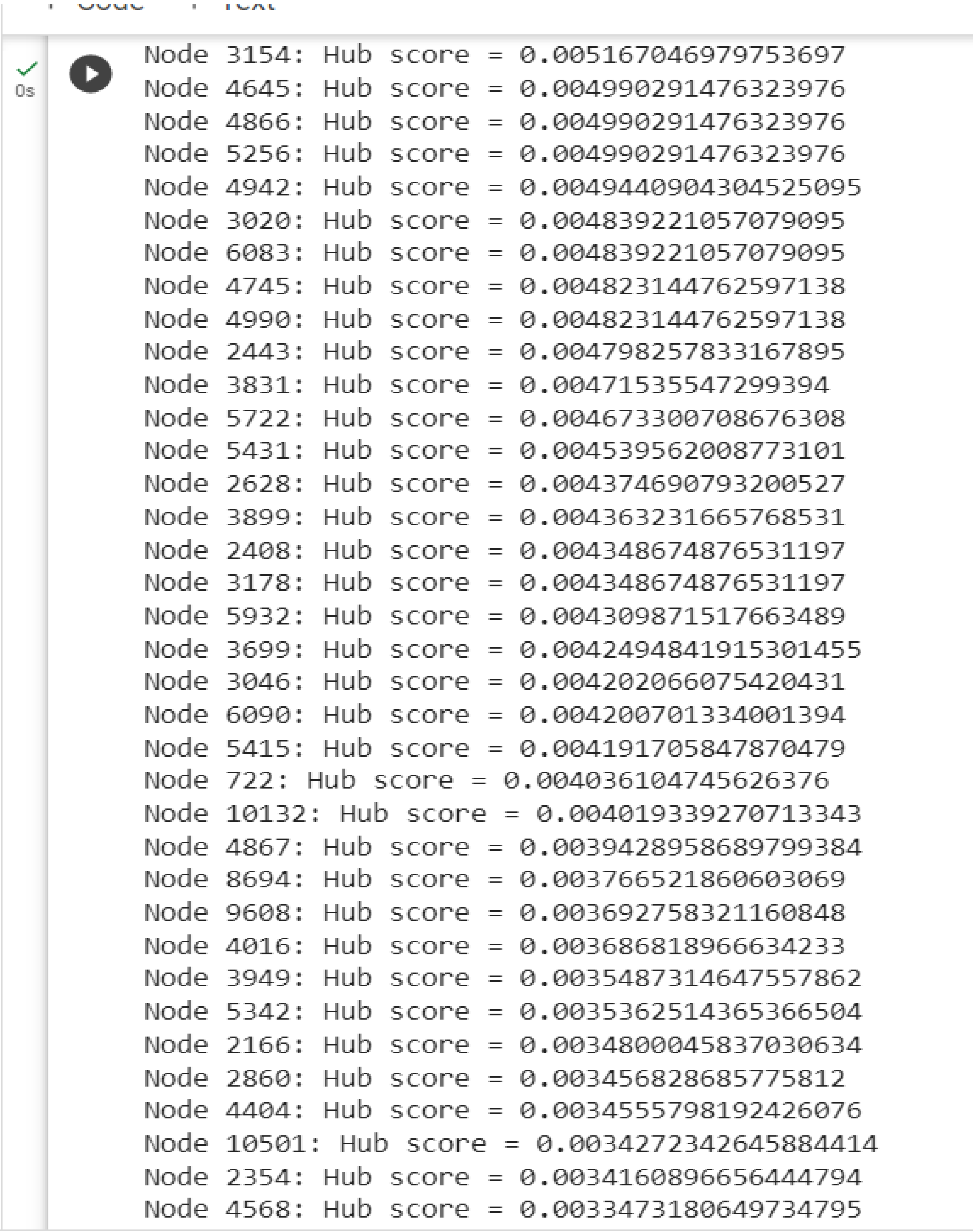
Page Rank is an algorithm that ranks web pages based on their relevance and returns them in order. Pages with more incoming edges are assigned a higher Page Rank score.



```
10876
{'1056': 0.0006711727183638692,
 '1054': 0.0006625823159671494,
 '1536': 0.0005496656851576059,
 '171': 0.0005434801433858492,
 '453': 0.0005243733925984653,
 '407': 0.0005097076151434906,
 '263': 0.0005079084313868783,
 '4664': 0.0005023514218978612,
 '1959': 0.0004892066518182482,
 '261': 0.0004858173126082881,
 '410': 0.0004849712292876587 4,
 '165': 0.0004841257759138581,
 '1198': 0.0004610584770426449,
 '127': 0.0004487289379563434 7,
 '4054': 0.000437794140953553,
 '2265': 0.00043229619669039003,
 '345': 0.0004310608487193007 7,
 '763': 0.000430458145624181 8,
 '989': 0.0004209253446550771 6,
 '987': 0.000418279176511429,
 '408': 0.0004151755498707455,
 '329': 0.0004112150077450631 5,
 '903': 0.000407752743756042 2,
 '4': 0.0004054552864779813 5,
 '1551': 0.0004007593566004388 4,
 '699': 0.0003985062208473963 3,
 '481': 0.000398215415564260 9,
 '1598': 0.0003969024535543202 7,
 '2485': 0.0003951036965313030 7,
 '982': 0.000393213896395912 4,
 '1055': 0.0003926498542443025 3,
 '2975': 0.000391724103875109 2,
```

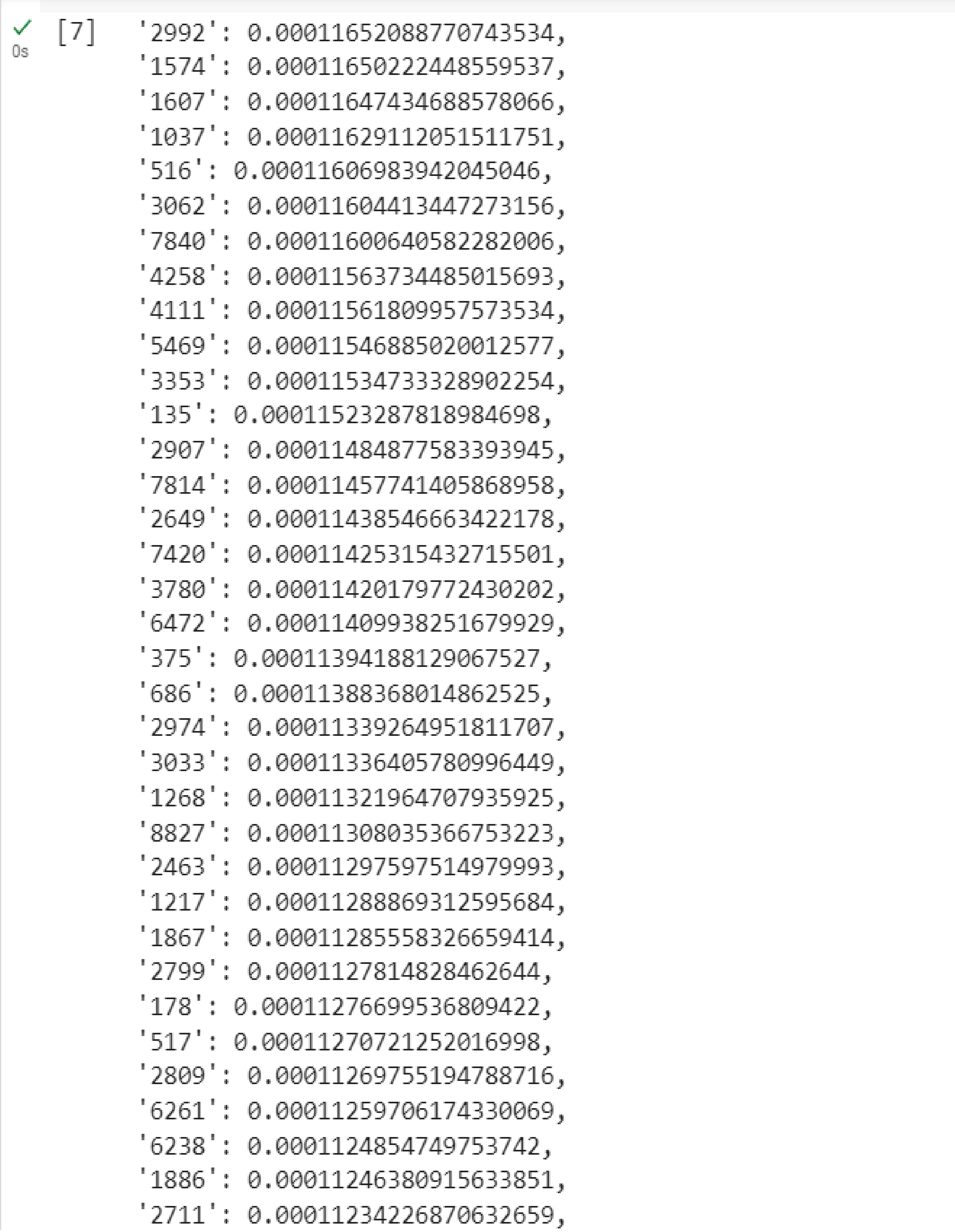
**Ans 2.2 : Hubs**

This method is used to measure the importance of web pages, where the root nodes are the web pages that are highly related to the provided query. Non-relevant pages that point to these root nodes are referred to as hubs. A good authority page will have many hubs pointing to it. Conversely, a page that many hubs link to is also considered important. The set of highly relevant web pages that are identified as roots are also known as potential authoritative pages.



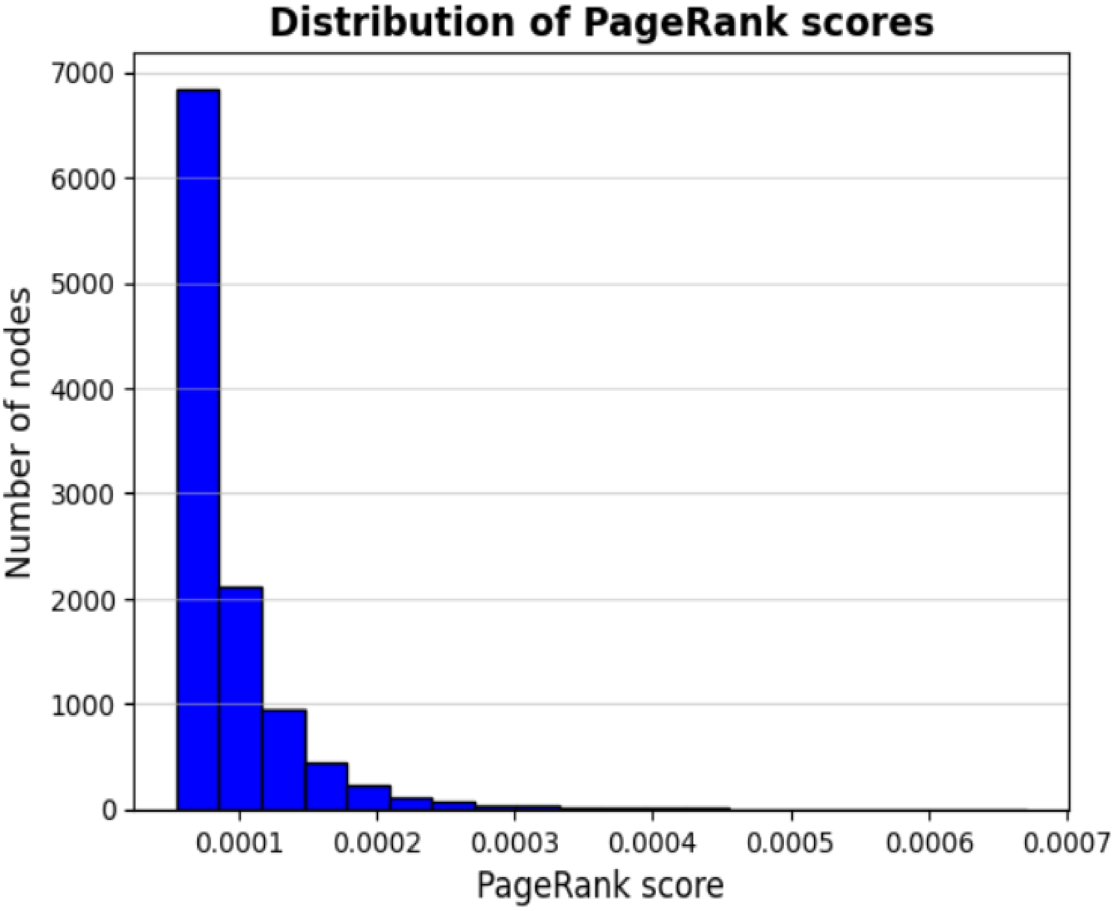
```
Node 3154: Hub score = 0.005167046979753697
Node 4645: Hub score = 0.004990291476323976
Node 4866: Hub score = 0.004990291476323976
Node 5256: Hub score = 0.004990291476323976
Node 4942: Hub score = 0.0049440904304525095
Node 3020: Hub score = 0.004839221057079095
Node 6083: Hub score = 0.004839221057079095
Node 4745: Hub score = 0.004823144762597138
Node 4990: Hub score = 0.004823144762597138
Node 2443: Hub score = 0.004798257833167895
Node 3831: Hub score = 0.00471535547299394
Node 5722: Hub score = 0.004673300708676308
Node 5431: Hub score = 0.004539562008773101
Node 2628: Hub score = 0.004374690793200527
Node 3899: Hub score = 0.004363231665768531
Node 2408: Hub score = 0.004348674876531197
Node 3178: Hub score = 0.004348674876531197
Node 5932: Hub score = 0.004309871517663489
Node 3699: Hub score = 0.0042494841915301455
Node 3046: Hub score = 0.004202066075420431
Node 6090: Hub score = 0.004200701334001394
Node 5415: Hub score = 0.004191705847870479
Node 722: Hub score = 0.004036104745626376
Node 10132: Hub score = 0.004019339270713343
Node 4867: Hub score = 0.0039428958689799384
Node 8694: Hub score = 0.003766521860603069
Node 9608: Hub score = 0.003692758321160848
Node 4016: Hub score = 0.003686818966634233
Node 3949: Hub score = 0.0035487314647557862
Node 5342: Hub score = 0.0035362514365366504
Node 2166: Hub score = 0.0034800045837030634
Node 2860: Hub score = 0.003456828685775812
Node 4404: Hub score = 0.0034555798192426076
Node 10501: Hub score = 0.0034272342645884414
Node 2354: Hub score = 0.0034160896656444794
Node 4568: Hub score = 0.0033473180649734795
```

Authority scores for all the nodes:

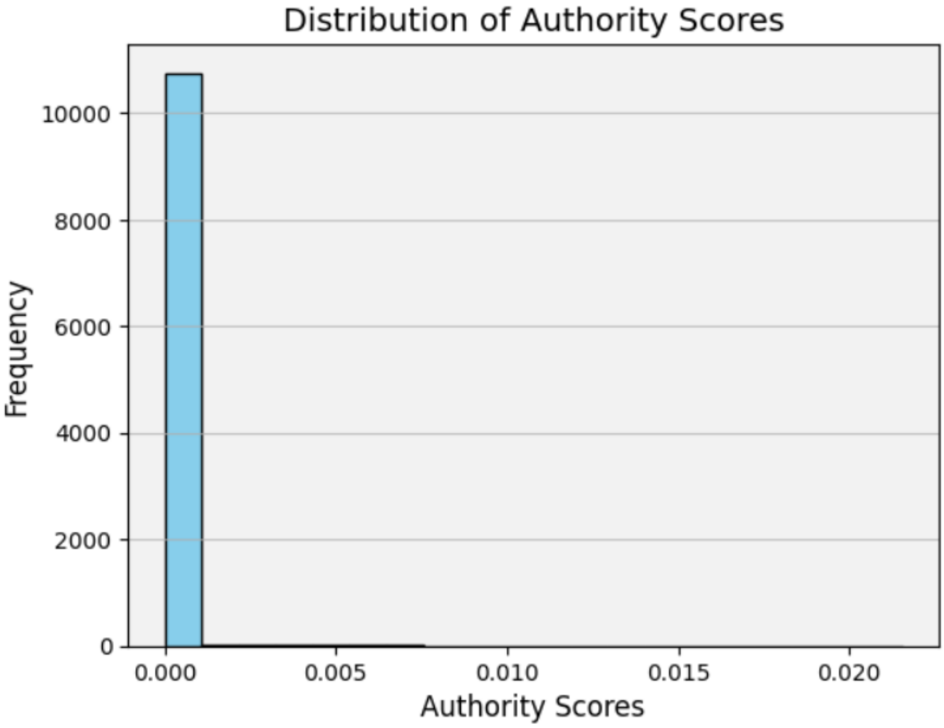


```
'2992': 0.00011652088770743534,
'1574': 0.00011650222448559537,
'1607': 0.00011647434688578066,
'1037': 0.00011629112051511751,
'516': 0.00011606983942045046,
'3062': 0.00011604413447273156,
'7840': 0.00011600640582282006,
'4258': 0.00011563734485015693,
'4111': 0.00011561809957573534,
'5469': 0.00011546885020012577,
'3353': 0.00011534733328902254,
'135': 0.00011523287818984698,
'2907': 0.00011484877583393945,
'7814': 0.00011457741405868958,
'2649': 0.00011438546663422178,
'7420': 0.00011425315432715501,
'3780': 0.00011420179772430202,
'6472': 0.00011409938251679929,
'375': 0.00011394188129067527,
'686': 0.00011388368014862525,
'2974': 0.00011339264951811707,
'3033': 0.00011336405780996449,
'1268': 0.00011321964707935925,
'8827': 0.00011308035366753223,
'2463': 0.00011297597514979993,
'1217': 0.00011288869312595684,
'1867': 0.00011285558326659414,
'2799': 0.00011278148284626644,
'178': 0.00011276699536809422,
'517': 0.00011270721252016998,
'2809': 0.00011269755194788716,
'6261': 0.00011259706174330069,
'6238': 0.00011248547497537342,
'1886': 0.00011246380915633851,
'2711': 0.00011234226870632659,
```

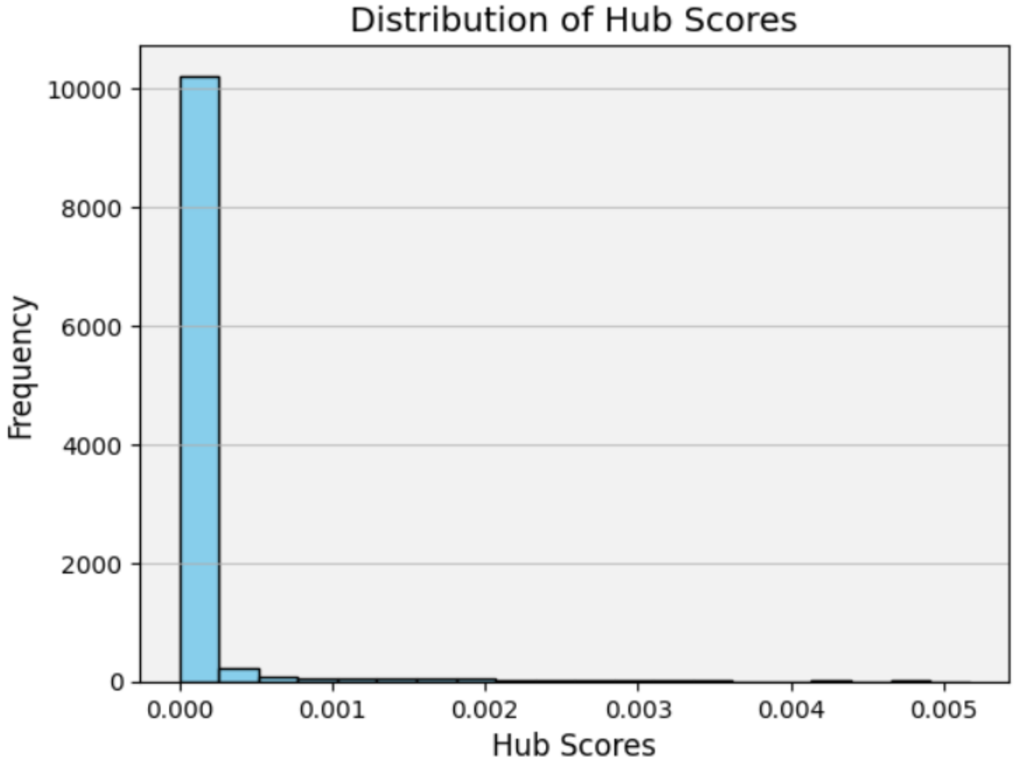
Distribution of page rank values :



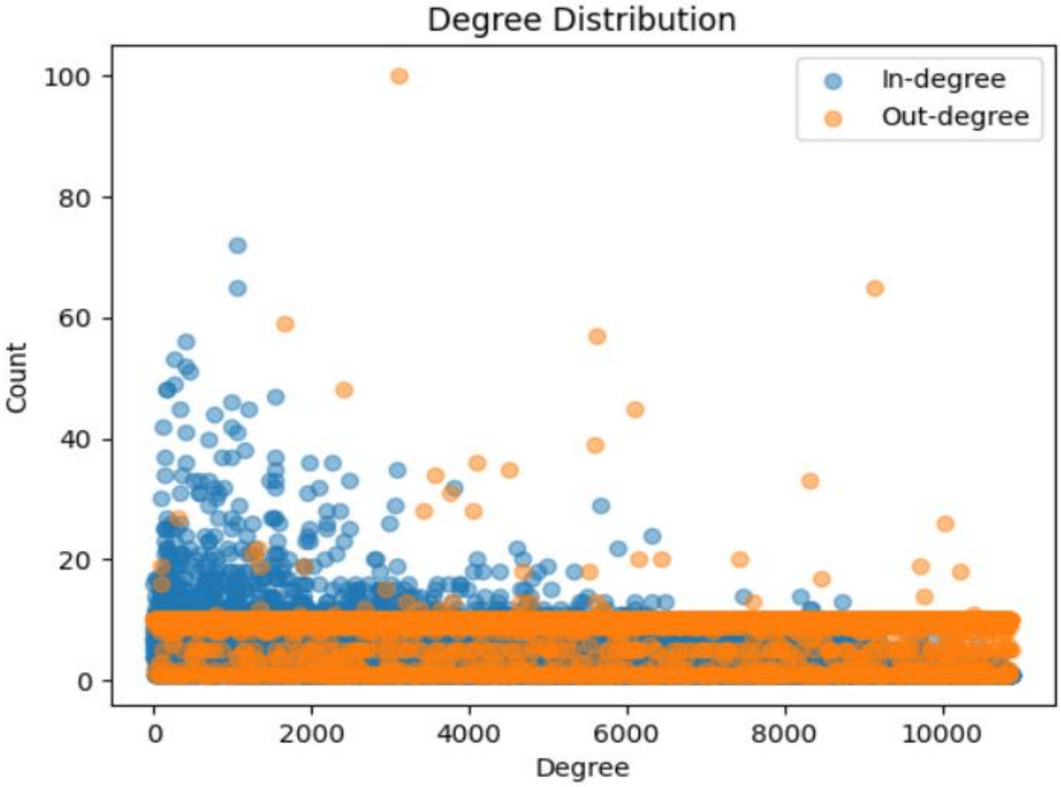
Distribution of Authority scores:



Distribution of Hub Score:

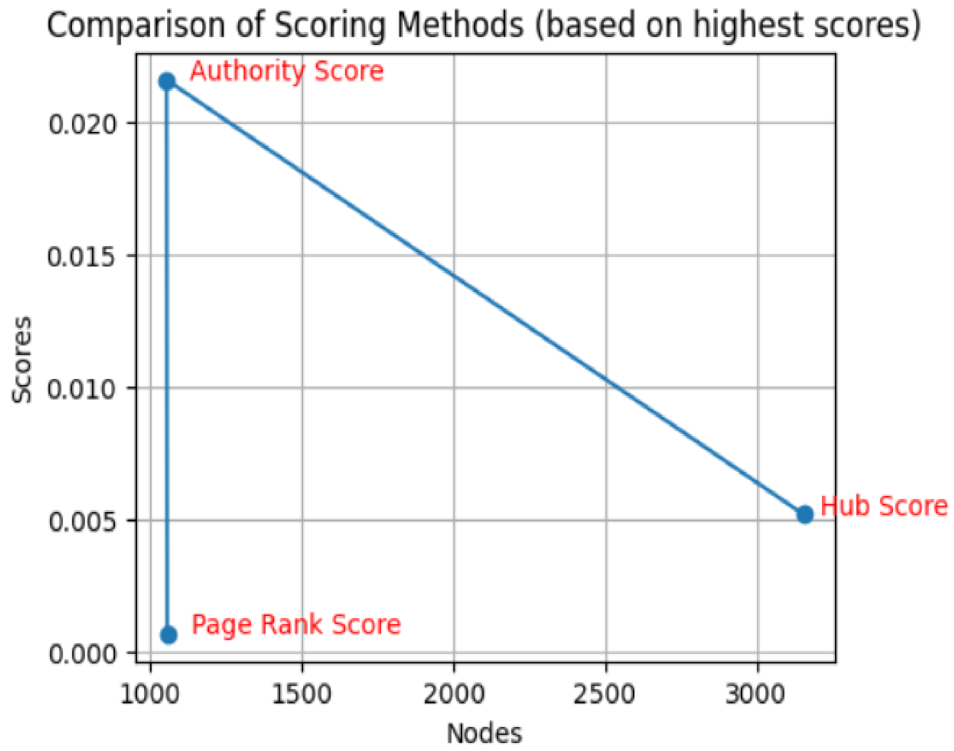


Comparison of all the values of all three scores using scatter plot

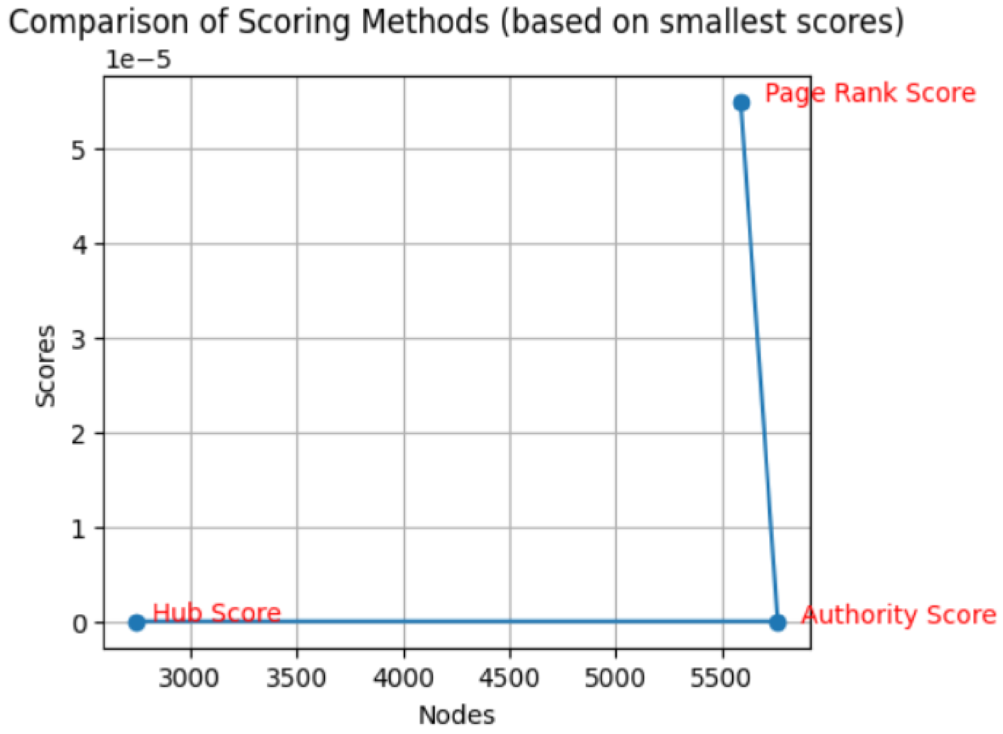


Comparison of all the three scores based on the highest score , average score and smallest score:

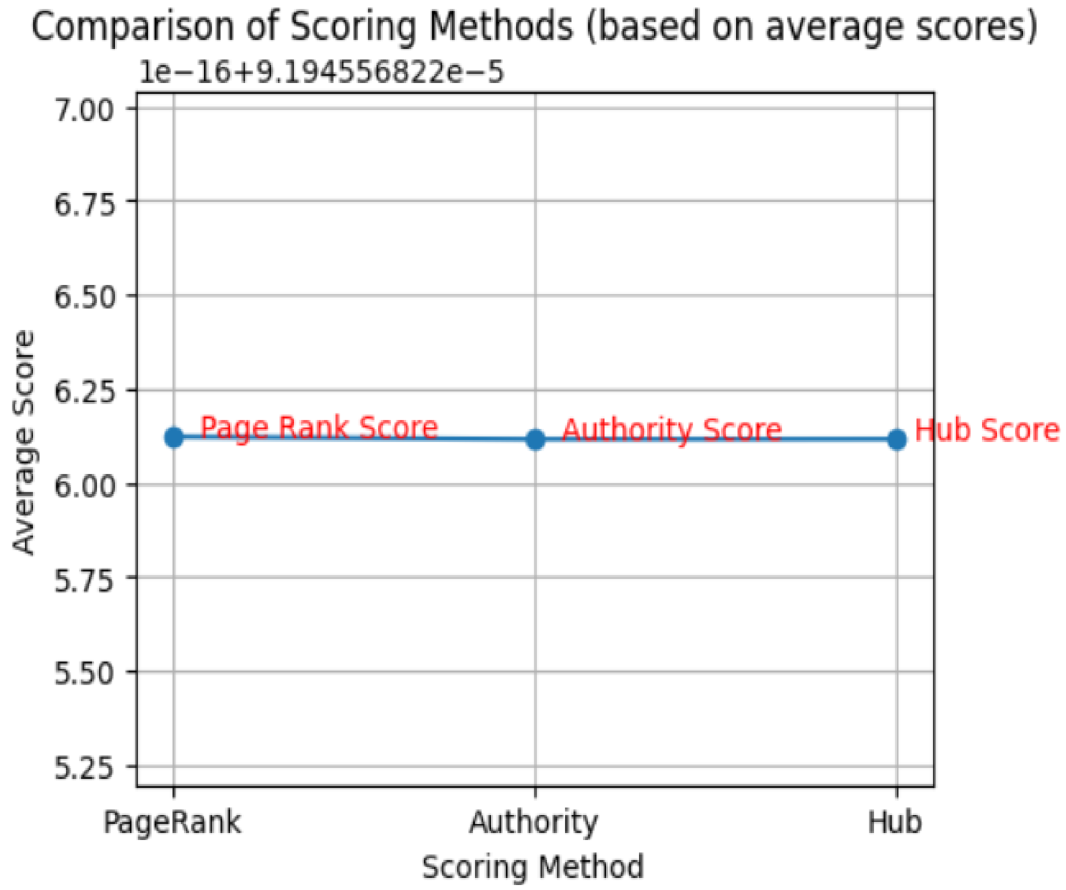
On highest score :



On smallest score:



On Average score:



The time required to evaluate scores in the HITS algorithm is typically greater than the time taken to evaluate scores using the PageRank algorithm. This is because HITS creates mutual reinforcement between authority and hub scores, whereas PageRank only considers authority. As a result, the HITS algorithm may yield less relevant results compared to PageRank.

PageRank's popularity is due to its efficiency, feasibility, and lower query time cost, among other features. These features are absent in the HITS algorithm, which may contribute to its lower popularity compared to PageRank.