IR-Assignment: 3

Devang Prajapati (MT22050)
Heli Shah (MT22100)

Dataset:

For our assignment we have selected the email-Eu-core network dataset. This network was generated by using e-mail data by a large European research institute.

If a person ‘u’ is sending an e-mail to person ‘v’ than there is an edge from ‘u’ to ‘v’. And the members of the institute are considered as nodes.

Dataset Statistics:

Dataset statistics	
Nodes	1005
Edges	25571
Nodes in largest WCC	986 (0.981)
Edges in largest WCC	25552 (0.999)
Nodes in largest SCC	803 (0.799)
Edges in largest SCC	24729 (0.967)
Average clustering coefficient	0.3994
Number of triangles	105461
Fraction of closed triangles	0.1085
Diameter (longest shortest path)	7
90-percentile effective diameter	2.9

No of Nodes: 1005
No of Edges: 25571
Type of Graph: Directed Graph.

Adjacency Matrix:

We have created the matrix of size “V x V” where V is the total number of nodes in the dataset.

We have filled that matrix with ‘0’ initially.

After that by traversing all the nodes we will add the value ‘+1’ to that place.

```
array([[1., 1., 0., ..., 0., 0., 0.],
       [0., 1., 0., ..., 0., 0., 0.],
       [0., 0., 1., ..., 0., 0., 0.],
       ...,
       [0., 0., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.],
       [0., 0., 0., ..., 0., 0., 0.]])
```

Edge List Creation:

We have made list of lists for the edge list.

We have traversed the file “converted.txt” and added the list [int(sender), int(receiver)] in an edge list.

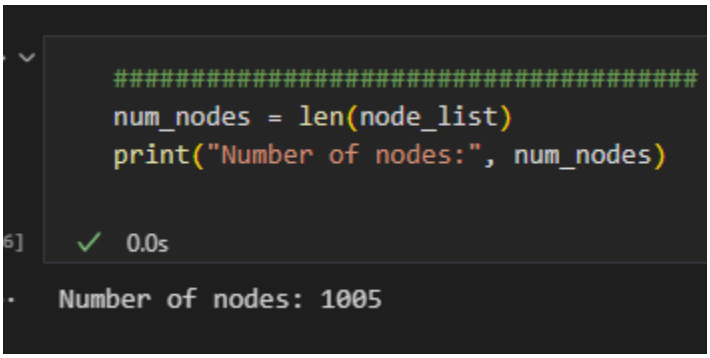
Final length of edge list would be total number of edges in the dataset.

```
[[1. 1. 0. ... 0. 0. 0.]
 [0. 1. 0. ... 0. 0. 0.]
 [0. 0. 1. ... 0. 0. 0.]
 ...
 [0. 0. 0. ... 0. 0. 0.]
 [0. 0. 0. ... 0. 0. 0.]
 [0. 0. 0. ... 0. 0. 0.]]
[[0, 1], [2, 3], [2, 4], [5, 6], [5, 7], [8, 9], [10, 11], [12, 13],
```

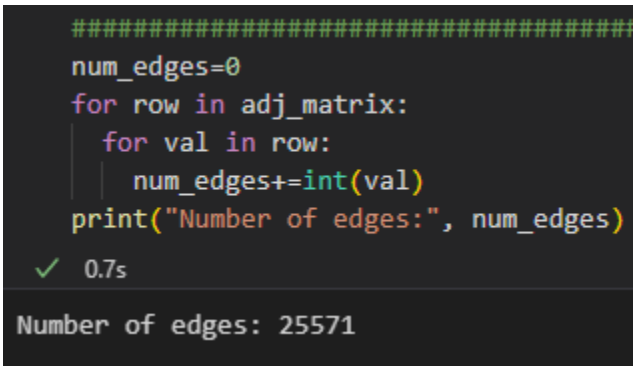
Nodes:

We have created a set and then traverse through the file. We added all the senders and receivers to the set, though the set will take only unique values.

We have converted that set to list.
We will get the number of nodes by printing the length of generated list.



Edges:
To get the number of edges we directly print the length of edge list.



Average In-degree and Maximum In-degree:
For calculating sum of in-degrees we will add the values of the columns from adjacency list.
Sum of each column represents in-degree of that node.
Column having maximum sum is the node having maximum in-degree.

```
Average in-degree: 25.443781094527363
Maximum in-degree: 212.0
Node with maximum in-degree: 160
```

Average Out-degree and Maximum Out-degree:

For calculating sum of out-degrees we will add the values of the rows from adjacency list.

Sum of each row represents out-degree of that node.

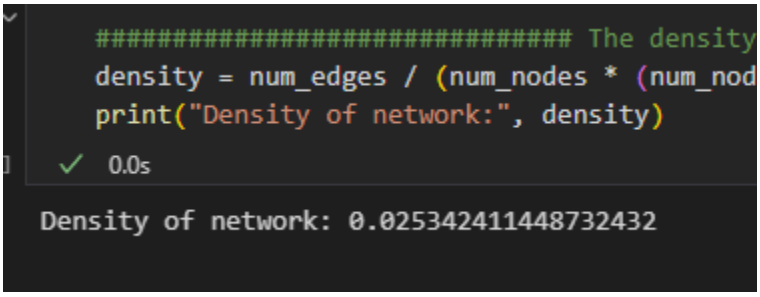
Row having maximum sum is the node having maximum out-degree.

```
Average out-degree: 25.443781094527363
Maximum out-degree: 334.0
Node with maximum out-degree: 160
```

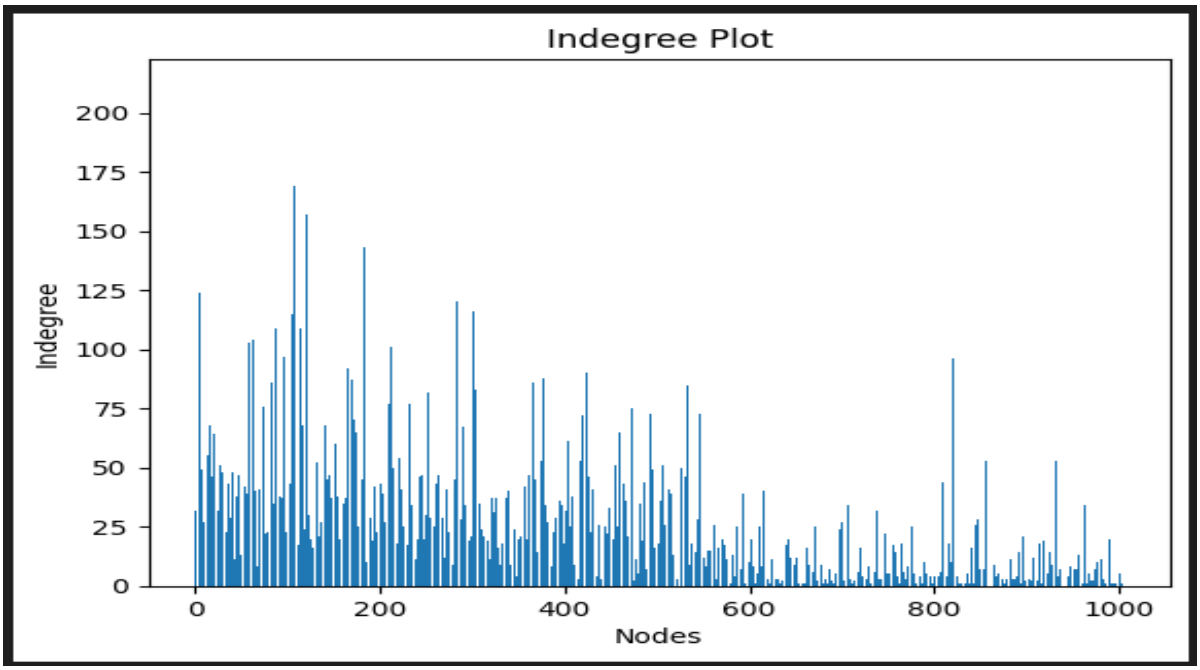
Network Density:

Density of the network is calculated by number of edges in graph divided by total number of edges possible in that graph.

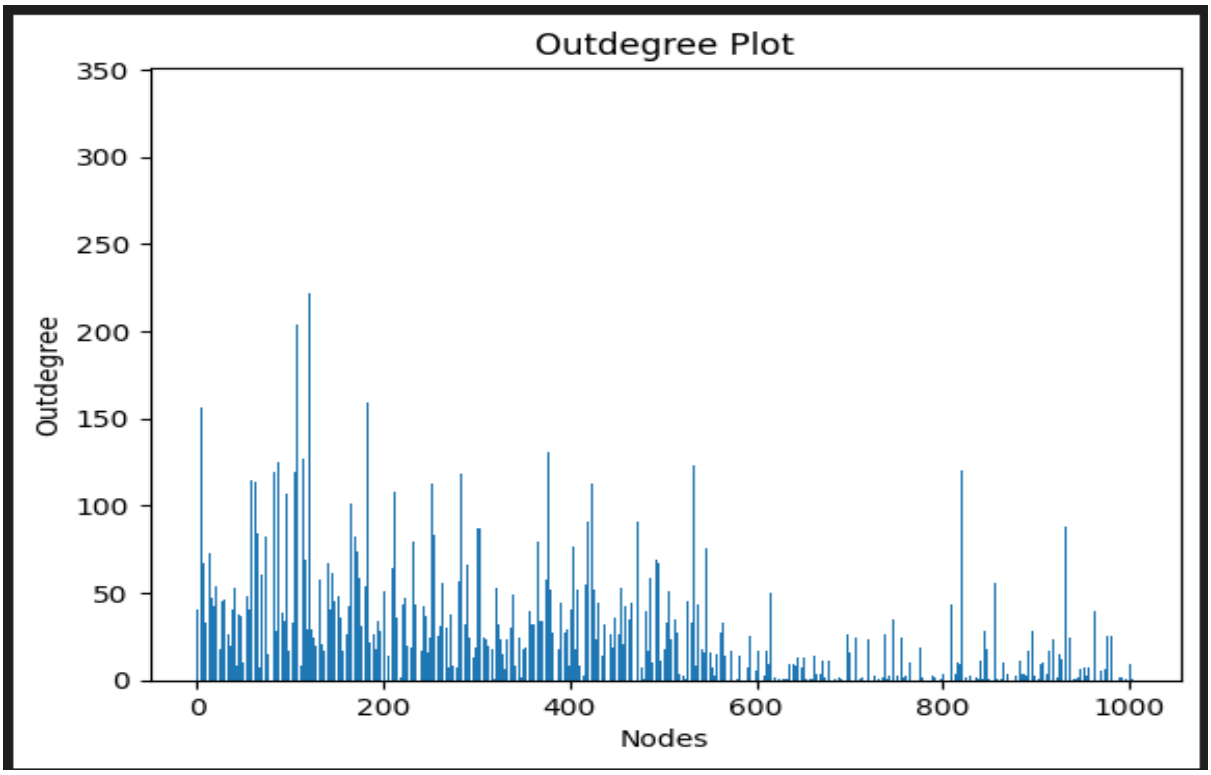
Density of our network is 0.025 approximately.



In-degree Plot:



Out-degree Plot:



Local Clustering Co-efficient:

The local Clustering Coefficient for a directed graph can be calculated as:

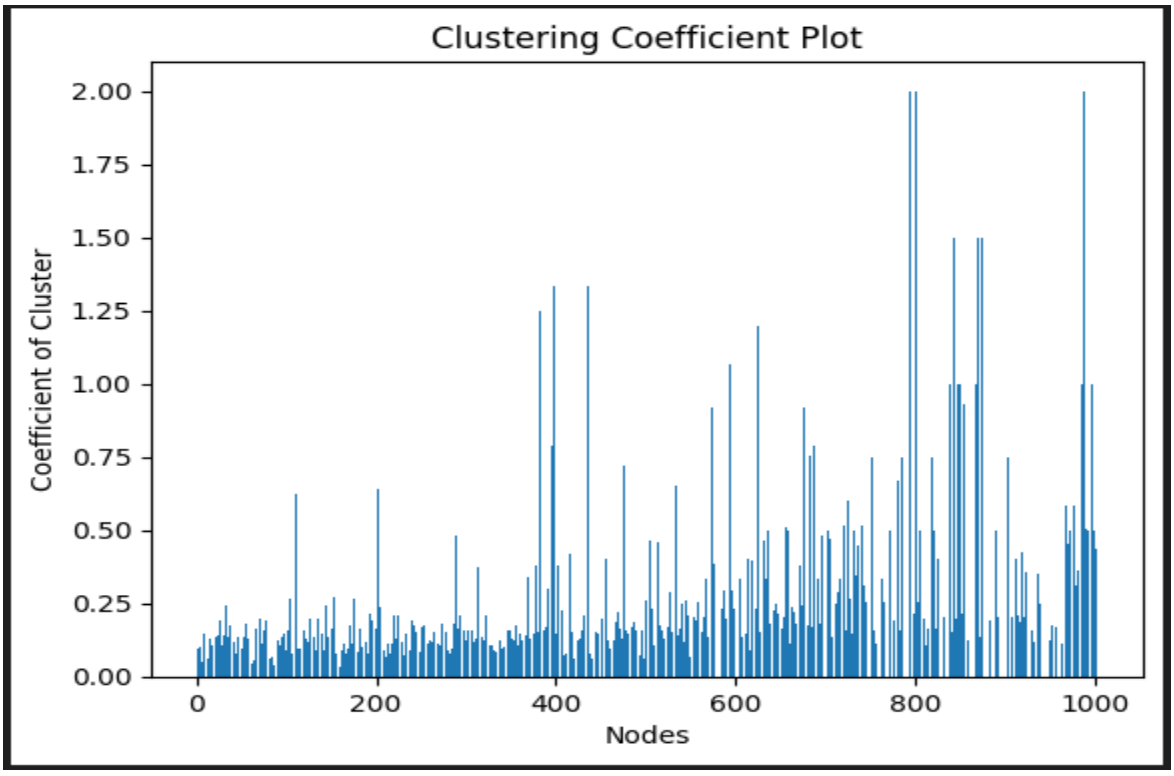
Local clustering coefficient for a node = n /(k * (k-1))

Where k represents the no of neighbors of a node and n represents the no of edges that are present in the graph.

Implementation:

- For each node in the graph, find all of its neighbors. A neighbor is any node that is directly connected to the node in question by an edge. For execution of the above concept, we have implemented a “noOfNeighbour” function.
- Then we calculated the number of edges between the neighbors of the node and divided by maximum number of possible edges between the neighbors of the node which is (k * (k-1)).
- Finally, implemented above two steps for all the nodes.

Clustering co-efficient plot:



Question: 2

We have calculated pagerank score, authority score and hub score for each node in this question.
We have used “networkx” module to calculate all the scores.
After that we have compared all the 3 methods which calculates the score.
We are getting score of ‘**node 160’** as the maximum score.
Page Rank scores of each node:

```
Output exceeds the size limit. O
{'0': 0.0013254022611944609,
 '1': 0.0047424807376661195,
 '2': 0.0021487010662397973,
 '3': 0.0017451782688328115,
 '4': 0.0024742123935726722,
 '5': 0.004939870122029279,
 '6': 0.003027584513968539,
 '7': 0.0018649686629997378,
 '8': 0.0012191439592595305,
 '9': 0.0011792012868826742,
 '10': 0.0013787386159000493,
 '11': 0.0023089700898852043,
 '12': 0.001644777292733341,
 '13': 0.002218675245615538,
 '14': 0.0017488834780439453,
 '15': 0.0016397542248842737,
```

Hit score of all nodes in the graph:

```
Output exceeds the size limit. Op
({'0': 0.001165652358166102,
  '1': 2.900462290263985e-05,
  '2': 0.003125339506305259,
  '3': 0.002490764619390502,
  '4': 0.0038790281781767,
  '5': 0.005048254863291742,
  '6': 0.003451142063154338,
  '7': 0.001467026939246893,
  '8': 0.0007134118810922479,
  '9': 0.0003877862894847761,
  '10': 0.0016319219935539 18,
  '11': 0.0020545042180840876,
  '12': 0.0014657908570417644,
  '13': 0.005345007274275857,
  '14': 0.0022753271340566384,
  '15': 0.0014345706495043505,
```

Authority score of each node:

```
Output exceeds the size limit. Op
{'0': 0.0008525503876603128,
 '1': 0.001702406148962447,
 '2': 0.002778761223017841,
 '3': 0.0027238509491243706,
 '4': 0.00318600346159355,
 '5': 0.003917430684266548,
 '6': 0.002512986894272628,
 '7': 0.0010488209777385633,
 '8': 0.0009345511864358768,
 '9': 0.000288495995933625,
 '10': 0.0018150659893945184,
 '11': 0.0015419432490675982,
 '12': 0.00167866298173424,
 '13': 0.0024254765082370273,
 '14': 0.00163706126775928,
 '15': 0.0015378836679498097,
```

Hub score of each node:

```
Output exceeds the size limit. Op
{'0': 0.0011656523581661006,
 '1': 2.900462290263982e-05,
 '2': 0.003125339506305257,
 '3': 0.002490764619390501,
 '4': 0.0038790281781766983,
 '5': 0.0050482548632917405,
 '6': 0.003451142063154335,
 '7': 0.001467026939246893,
 '8': 0.0007134118810922473,
 '9': 0.0003877862894847756,
 '10': 0.0016319219935539183,
 '11': 0.0020545042180840876,
 '12': 0.0014657908570417644,
 '13': 0.005345007274275859,
 '14': 0.0022753271340566384,
 '15': 0.0014345706495043516,
```

Comparison of PageRank, Authority and Hub Score:

Top 10 nodes having maximum score using PageRank algorithm:

Node 160 has 0.0069 score approximately using PageRank algorithm and so on.

```
160: 0.0069070158051178785
62: 0.005413484434927099
86: 0.00532331046872205
107: 0.00513991230347384
5: 0.004939870122029279
121: 0.004749279211807127
1: 0.0047424807376661195
129: 0.0046405224812937325
365: 0.004403126469281699
64: 0.0044000938826829735
```

Top 10 nodes having maximum score using HITS (Authority Score) algorithm:

Node 160 has 0.0072 score approximately using HITS (Authority Score) algorithm and so on.

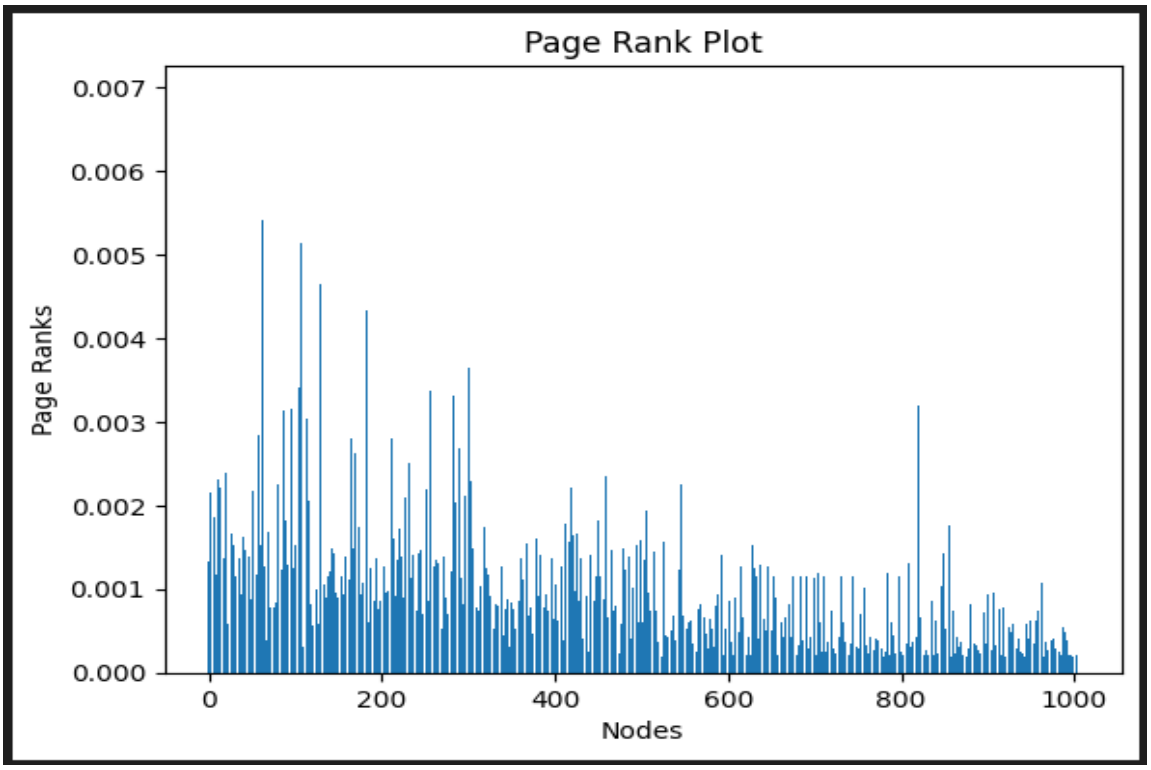
```
160: 0.007220481699191956
107: 0.006898170199864647
62: 0.006695883147202667
434: 0.006485092543979912
121: 0.006471582443168834
183: 0.006040848953683613
128: 0.005947949784933278
249: 0.005729100052968212
256: 0.005703873069050456
129: 0.005677728282821713
```

Top 10 nodes having maximum score using HITS (Hub Score) algorithm:

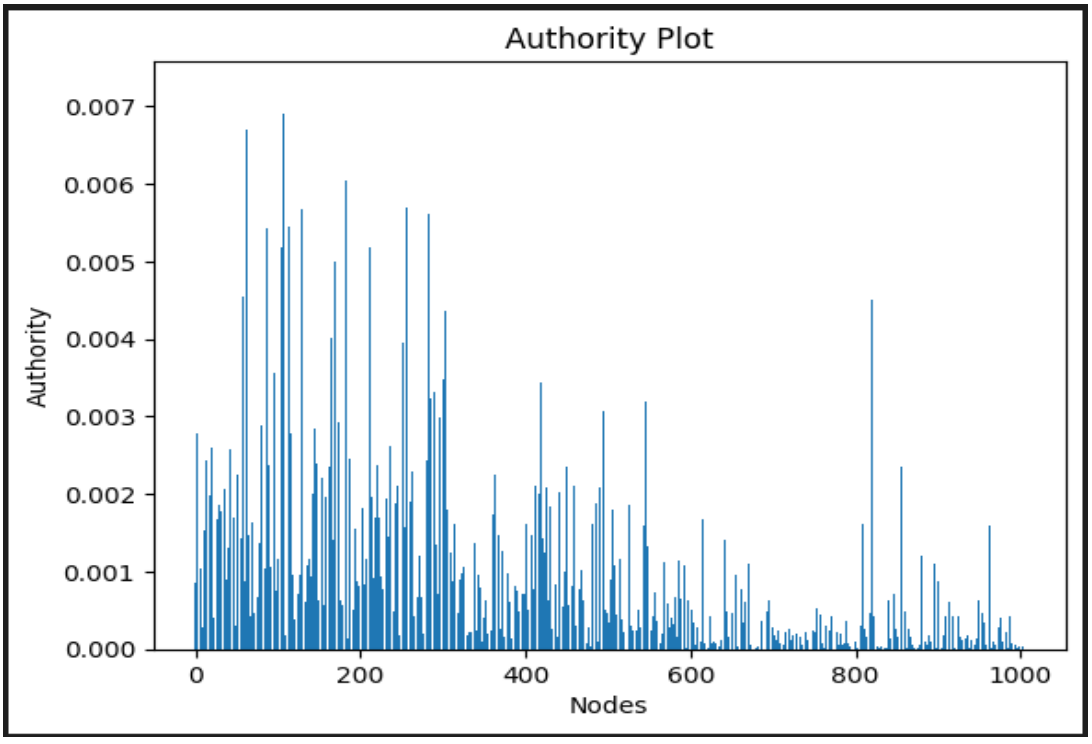
Node 160 has 0.0106 score approximately using HITS (Hub Score) algorithm and so on.

```
160: 0.01062880261103844
82: 0.009616665861905403
121: 0.009530349046577478
107: 0.00878806711376408
62: 0.008232597715453004
249: 0.008017503203921339
434: 0.0075412520505521155
183: 0.0072000203311382875
86: 0.00700307416176555
114: 0.006398316126418318
```

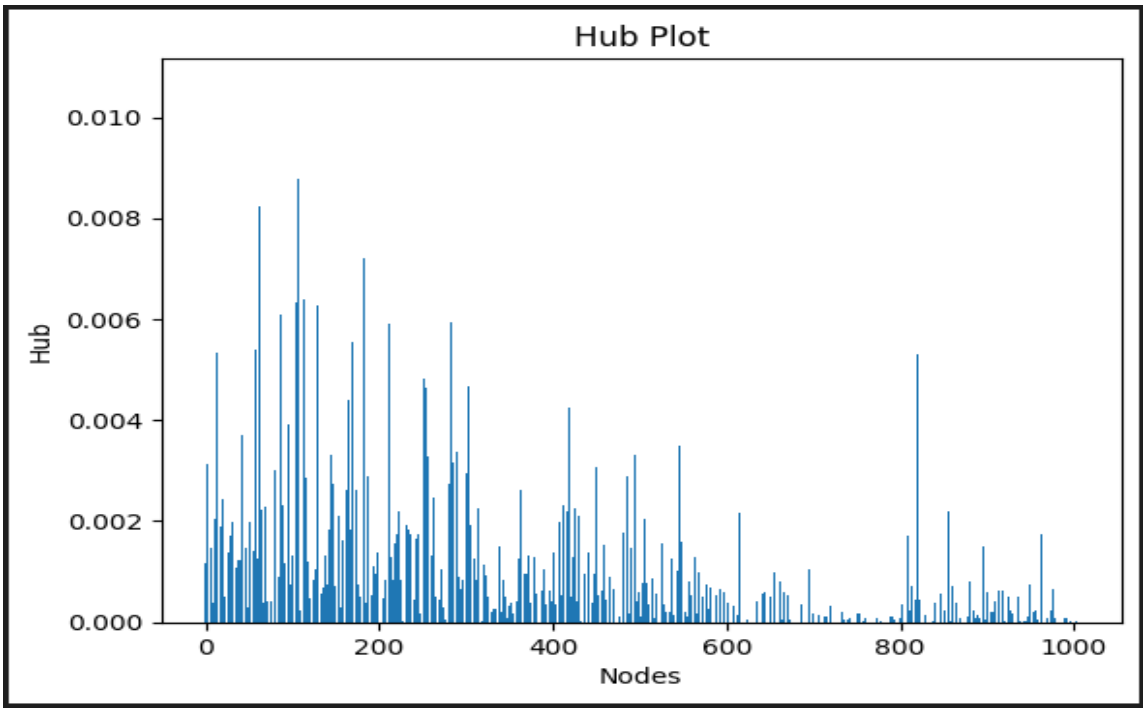
Plot of PageRank scores of all nodes:



Plot of Authority scores of all nodes:



Plot of hub scores of all nodes:



Scores of all the nodes using all 3 algorithms:

Node	PageRank	Authority	Hub
0	0.001325	0.000853	0.001166
1	0.004742	0.001702	0.000029
2	0.002149	0.002779	0.003125
3	0.001745	0.002724	0.002491
4	0.002474	0.003186	0.003879
...	...	...	...
1000	0.000328	0.000144	0.000207
1001	0.000310	0.000189	0.000234
1002	0.000202	0.000022	-0.000000
1003	0.000204	0.000030	0.000025
1004	0.000207	0.000020	-0.000000

[1005 rows x 3 columns]

It’s pretty evident from the above snapshots that the PageRank algorithm and HITS algorithm rank node 160 as the highest among all 1005 nodes in the entire dataset graph.
That means node 160 will have more importance than others in the graph.
Here nodes like 160,62,86,107,121 are the nodes that are coming on to the top using both the scoring technique.
If we deeply compare both the algorithms in the terms of absolutely pristine, the relevancy of the PageRank algorithm is more compared to the HITS algorithm.