WORLD CITIES AIRLINE NETWORK ANALYSIS USING CENTRALITY MEASURES

A PROJECT REPORT

Submitted by

M TANUJ	**BL.EN.U4AIE19041**
P AKHIL	**BL.EN.U4AIE19049**
V AISHWARYA	**BL.EN.U4AIE19068**

for the course

19AIE203- Data Structures and Algorithms – 2

Guided and Evaluated by

D. RADHA

Asst. Prof(SG),

Dept. of CSE,

AMRITA SCHOOL OF ENGINEERING, BANGALORE

AMRITA VISHWA VIDHYAPEETHAM

BANGALORE-560 035

December-2020

ABSTRACT OF THE PROJECT

The air transportation industry has a great impact on the economy. In this, we analyze the air transportation network in the world to better understand its characteristics. The world's major airports are directly linked to hundreds of airports without intermediate routes. This connection can be described as the network in which the airport is a node and the route will be a connection line. Here we analyze the network which contains 67663 routes between 3321 airports on 548 airlines spanning the globe. We will find the most influential airport in the world using centrality measures such as betweenness centrality, degree centrality, closeness centrality, eigen vector centrality and data analysis. Some of the characteristics are like the busiest airport and the airports which influence trade, alternate path, fastest route, nearest airports, etc. The characteristics helps to find the designated airports meant for improving the economy. The results of this project say about the prominent communication and connections among the airports worldwide.

INTRODUCTION

Transport network analysis is used to determine the flow of vehicles (or people) through a transport network, typically using mathematical graph theory. It may combine different modes of transport, for example, walking and car, to model multi-modal journeys. Transport network analysis falls within the field of transport engineering.

The term network refers to the framework of routes within a system of locations, identified as nodes. A route is a single link between two nodes that are part of a larger network that can refer to tangible routes such as roads and rails, or less tangible routes such as air and sea corridors.

Therefore, graphs are the spatial representation of airline networks. Graphs are useful because they represent graphically the airline networks and above all provide a formal representation of them. Moreover, an index can be calculated to describe specific structural dimensions, such as density, inclusion and cohesion.

Influential Network Analysis

Influential network analysis enables you to take a closer look at the positions of different users on a social networking site. It uses algorithms such as degree centrality and closeness centrality to indicate the impact each airport has in the network. In essence Network Analysis views relationships as links. Some people in the network might have only one or two connections, and others might have hundreds or thousands.

Centrality

In graph theory and network analysis, indicators of centrality identify the most important vertices within a graph. Applications include identifying the most influential person(s) in a social network, key infrastructure nodes in the Internet or urban networks, and super-spreaders of illness. Centrality concepts were first developed in social network analysis, and many of the terms used to measure centrality reflect their sociological origin. They should not be confused with node influence metrics, which seek to quantify the influence of every node in the network.

1. Betweenness Centrality

The betweenness centrality captures how much a given node (hereby denoted u) is in-between others. This metric is measured with the number of shortest paths (between any couple of nodes in the graphs) that passes through the target node *u.* This score is moderated by the total number of shortest paths existing between any couple of nodes of the graph. The target node would have a high betweenness centrality if it appears in many shortest paths. The betweenness centrality for each vertex is the number of these shortest paths that pass through the vertex.

The betweenness centrality of a node *u* is given by the expression:

$$g(v) = \sum_{s \neq v \neq t} \frac{\sigma_{st}(v)}{\sigma_{st}}$$

Where Sigma st is the total number of shortest paths from node s to node t

2. Closeness Centrality

Closeness centrality is a way of finding nodes that are able to spread data very efficiently through a graph. The closeness centrality of a node measures its average inverse distance to all other nodes. Nodes with a more closeness score have the shortest distances to all other nodes. Closeness was defined as the reciprocal of the farness

Closeness centrality will be given by

$$C(x) = \frac{1}{\sum_y d(y,x)}.$$

Where d(y,x) is the distance between the vertices y and x

3. Degree Centrality:

Degree centrality is defined as the number of links incident upon a node (i.e., the number of ties that a node has). If the network is directed (meaning that ties have direction), then two separate measures of degree centrality are defined, namely, indegree and outdegree. Indegree is a count of the number of ties directed to the node (head endpoints) and outdegree is the number of ties that the node directs to others (tail endpoints). In such cases, the degree is the sum of indegree and outdegree.

The degree centrality of a vertex *u,* for a given graph G= (V,E) where v is the vertices and E are the edges

$$C_D(v) = \deg(v)$$

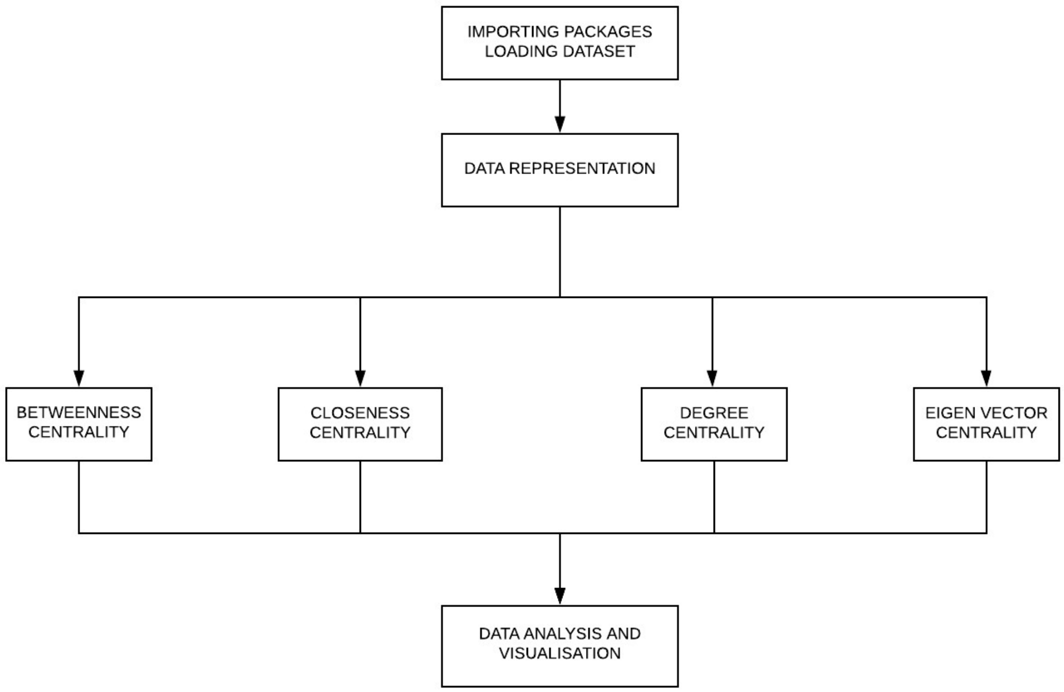
4. Eigen Vector Centrality:

Eigenvector centrality (also called eigen centrality) is a measure of the influence of a node in a network. It assigns relative scores to all nodes in the network based on the notion that connections to high-scoring nodes contribute more to the node score than equal connections to low-scoring nodes. Google's PageRank and the Katz centrality are variants of the eigenvector centrality

For a given graph G= (V, E) where v is the vertices and E are the edges A is a(v,t) is the adjacency matrix

$$x_v = \frac{1}{\lambda} \sum_{t \in M(v)} x_t = \frac{1}{\lambda} \sum_{t \in G} a_{v,t} x_t$$

SYSTEM MODEL – DESCRIPTION OF THE DIAGRAM



- Importing the packages and loading the dataset.
- Resenting the data using the graph.
- Calculating the centrality measures such as Betweenness centrality, Closeness Centrality, Degree Centrality, Eigen Vector Centrality
- Finally, analyzing the data and then visualizing it.

IMPLEMENTATION DETAILS

1. We have done the Airline Network Analysis in python using Google collab.

2. These are the datasets we have used:
   https://www.kaggle.com/divyanshrai/openflights-airports-database-2017
   https://www.kaggle.com/divyanshrai/openflights-route-database-2014

3. Packages used:
   - pandas: *pandas* is a fast, powerful, flexible and easy to use open-source data analysis and manipulation tool.

   - networkx: *NetworkX* is a Python package for the creation, manipulation, and study of the structure, dynamics, and functions of complex networks

   - matplotlib.pyplot: *Matplotlib* is a comprehensive library for creating static, animated, and interactive visualizations in Python.

   - numpy: *Numpy* is a python library which has functions for working in domain of linear algebra, fourier transform, and matrices.

   - seaborn: *Seaborn* is a Python data visualization library based on matplotlib. It provides a high-level interface for drawing attractive and informative statistical graphics.

4. Functions used:
   - nx.spring_layout(): Gives out a dictionary containing the position nodes from the graph.
   - nx.betweenness_centrality(): Computes the betweenness centrality of every node in a graph, and it returns a dictionary where the keys are the nodes and the values are their betweenness centrality measures.
   - nx.closeness_centrality() : Computes the closeness centrality of every node in a graph, and it returns a dictionary where the keys are the nodes and the values are their betweenness centrality measures.
   - nx.degree_centrality() : Computes the degree centrality of every node in a graph, and it returns a dictionary where the keys are the nodes and the values are their betweenness centrality measures.
   - nx.eigenvector_centrality(): Computes the eigenvector centrality of every node in a graph, and it returns a dictionary where the keys are the nodes and the values are their betweenness centrality measures.

SAMPLE CODE

BETWEENESS CENTRALITY

```
[158] pos = nx.spring_layout(GRAPH)
      betCent = nx.betweenness_centrality(GRAPH)
      node_color = [20000.0 * GRAPH.degree(v) for v in GRAPH]
      node_size =  [v * 50000 for v in betCent.values()]
      plt.figure(figsize=(20,12))
      nx.draw_networkx(GRAPH, pos=pos, with_labels=False,
                       node_color=node_color,
                       node_size=node_size )
      plt.axis('off')
```

CLOSENESS CENTRALITY

```
[159] pos = nx.spring_layout(GRAPH)
      cloCent = nx.closeness_centrality(GRAPH)
      node_color = [20000.0 * GRAPH.degree(v) for v in GRAPH]
      node_size =  [v * 500 for v in cloCent.values()]
      plt.figure(figsize=(20,12))
      nx.draw_networkx(GRAPH, pos=pos, with_labels=False,
                       node_color=node_color,
                       node_size=node_size )
      plt.axis('off')
```

DEGREE CENTRALITY

```
[162] pos = nx.spring_layout(GRAPH)
      degCent = nx.degree_centrality(GRAPH)
      node_color = [20000.0 * GRAPH.degree(v) for v in GRAPH]
      node_size =  [v * 50000 for v in degCent.values()]
      plt.figure(figsize=(20,12))
      nx.draw_networkx(GRAPH, pos=pos, with_labels=False,
                       node_color=node_color,
                       node_size=node_size )
      plt.axis('off')
```

EIGENVECTOR CENTRALITY

```
[163] pos = nx.spring_layout(GRAPH)
      eigCent = nx.eigenvector_centrality(GRAPH)
      node_color = [20000.0 * GRAPH.degree(v) for v in GRAPH]
      node_size =  [v * 50000 for v in eigCent.values()]
      plt.figure(figsize=(20,12))
      nx.draw_networkx(GRAPH, pos=pos, with_labels=False,
                       node_color=node_color,
                       node_size=node_size )
      plt.axis('off')
```

Top 5 airports of individual centralities

```
[235] df2 = pd.DataFrame()
      df2['betweeness_centrality'] = pd.Series(b1)
      df2['closeness_centrality'] = pd.Series(c1)
      df2['degree_centrality'] = pd.Series(d1)
      df2['eigenvector_centrality'] = pd.Series(e1)
      df2
```

```
airport_dfs = airport_df.copy()
airport_dfs["betweeness_centrality"]  =  airport_dfs["IATA"].map(betCent)
airport_dfs["closeness_centrality"]   =  airport_dfs["IATA"].map(cloCent)
airport_dfs["degree_centrality"]      =  airport_dfs["IATA"].map(degCent)
airport_dfs["eigenvector_centrality"] =  airport_dfs["IATA"].map(eigCent)
airport_dfs["mean_average"]           =  airport_dfs["IATA"].map(mean_value)
airport_dfs.head()
```

Top 20 Busiest Airports based on Mean Average Score

```
[178] airport_dfs1 = airport_dfs.sort_values(by=['mean_average'], ascending=False)
      airport_dfs2 = airport_dfs1.head(20)
      airport_dfs2
```

Pie Chart Representation of Top 20 Busiest Airports based on Mean Average

```
I = airport_dfs2["IATA"]
mean_avg1 = airport_dfs2["mean_average"]
fig = plt.figure(figsize =(10, 10))
plt.pie(mean_avg1,labels=I,autopct='%1.1f%%',startangle=90)
plt.title("TOP 20 BUSIEST AIRPORTS BASED ON MEAN AVERAGE VALUE")
plt.show()
```

Bar Chart Representation of Top 20 Busiest Airports based on Mean Average

```
airport_IATA =airport_dfs2['IATA']
mean_average1 = airport_dfs2['mean_average']
fig = plt.figure(figsize =(30, 7))

plt.bar(airport_IATA, mean_average1)

plt.xlabel("TOP 20 AIRPORTS")
plt.ylabel("MEAN AVERAGE SCORE")
plt.title("TOP 20 BUSIEST AIRPORTS BASED ON MEAN AVERAGE")
plt.show()
```

SAMPLE OUTPUT

Mapping Airports on world map:

Top 5 Airports based on Betweeness Centrality:

	Name	City	Country	IATA	Betweeness_centrality
3574	Ted Stevens Anchorage International Airport	Anchorage	United States	ANC	0.070204
3285	Los Angeles International Airport	Los Angeles	United States	LAX	0.066164
1346	Charles de Gaulle International Airport	Paris	France	CDG	0.061703
2100	Dubai International Airport	Dubai	United Arab Emirates	DXB	0.059350
336	Frankfurt am Main Airport	Frankfurt	Germany	FRA	0.051000

Top 5 Airports based on Closeness Centrality

:

	Name	City	Country	IATA	Closeness_centrality
336	Frankfurt am Main Airport	Frankfurt	Germany	FRA	0.392389
1346	Charles de Gaulle International Airport	Paris	France	CDG	0.389993
502	London Heathrow Airport	London	United Kingdom	LHR	0.388306
2100	Dubai International Airport	Dubai	United Arab Emirates	DXB	0.384084
574	Amsterdam Airport Schiphol	Amsterdam	Netherlands	AMS	0.382932

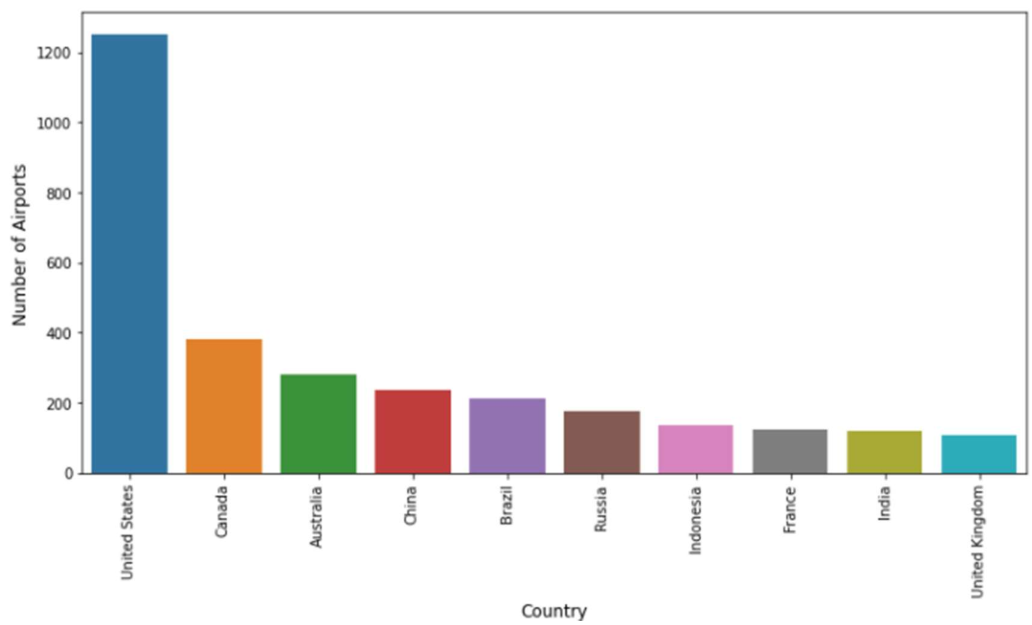
Top 5 Airports based on Degree Centrality:

	Name	City	Country	IATA	Degree_centrality
336	Frankfurt am Main Airport	Frankfurt	Germany	FRA	0.139311
1346	Charles de Gaulle International Airport	Paris	France	CDG	0.137266
574	Amsterdam Airport Schiphol	Amsterdam	Netherlands	AMS	0.135222
7629	Istanbul Airport	Istanbul	Turkey	IST	0.133470
3482	Hartsfield Jackson Atlanta International Airport	Atlanta	United States	ATL	0.126460

Top 5 Airports based on Eigenvector Centrality:

	Name	City	Country	IATA	Eigen_centrality
574	Amsterdam Airport Schiphol	Amsterdam	Netherlands	AMS	0.165909
336	Frankfurt am Main Airport	Frankfurt	Germany	FRA	0.165748
1346	Charles de Gaulle International Airport	Paris	France	CDG	0.159224
342	Munich Airport	Munich	Germany	MUC	0.148957
502	London Heathrow Airport	London	United Kingdom	LHR	0.137050

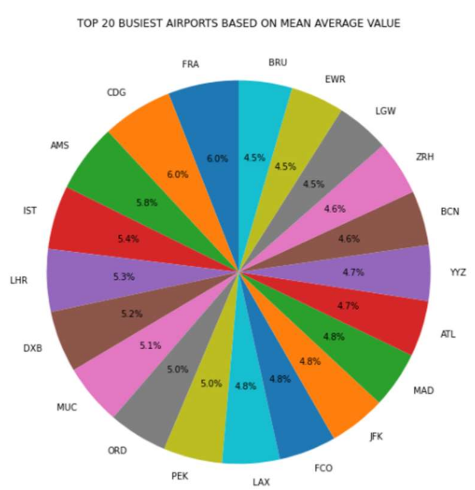
Top 10 Countries with Highest number of Airports:

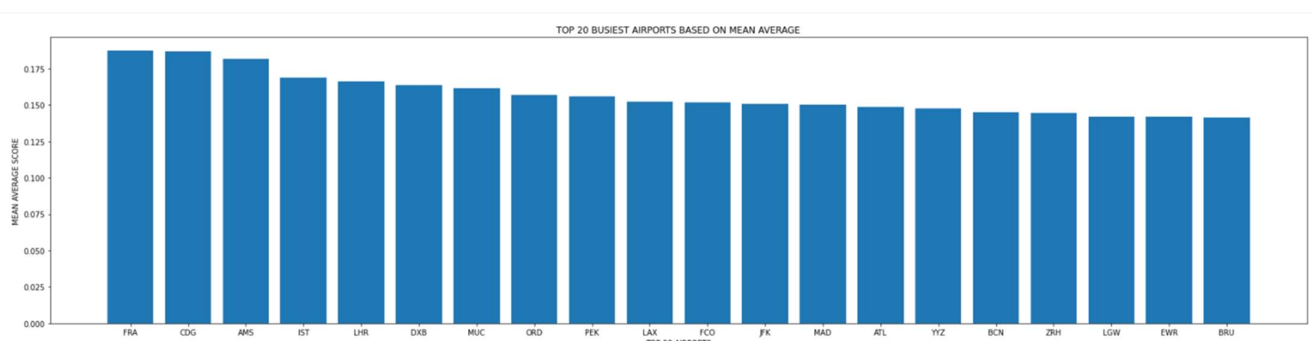


Top 20 Airports based on Mean Average Score of all Centralities:

	Name	City	Country	IATA	betweeness_centrality	closeness_centrality	degree_centrality	eigenvector_centrality	mean_average
336	Frankfurt am Main Airport	Frankfurt	Germany	FRA	0.051000	0.392389	0.139311	0.165748	0.187112
1346	Charles de Gaulle International Airport	Paris	France	CDG	0.061703	0.389993	0.137266	0.159224	0.187047
574	Amsterdam Airport Schiphol	Amsterdam	Netherlands	AMS	0.042658	0.382932	0.135222	0.165909	0.181680
7629	Istanbul Airport	Istanbul	Turkey	IST	0.041219	0.371370	0.133470	0.129644	0.168926
502	London Heathrow Airport	London	United Kingdom	LHR	0.038385	0.388306	0.099883	0.137050	0.165906
2100	Dubai International Airport	Dubai	United Arab Emirates	DXB	0.059350	0.384084	0.108061	0.102745	0.163560
342	Munich Airport	Munich	Germany	MUC	0.015418	0.370252	0.110981	0.148957	0.161402
3630	Chicago O'Hare International Airport	Chicago	United States	ORD	0.047430	0.370376	0.119451	0.090182	0.156860
3170	Beijing Capital International Airport	Beijing	China	PEK	0.049167	0.369593	0.120327	0.083801	0.155722
3285	Los Angeles International Airport	Los Angeles	United States	LAX	0.066164	0.379042	0.086741	0.075881	0.151957
1514	Leonardo da Vinci–Fiumicino Airport	Rome	Italy	FCO	0.012168	0.367100	0.092290	0.135864	0.151855
3597	John F Kennedy International Airport	New York	United States	JFK	0.025764	0.377277	0.094042	0.106159	0.150810
1196	Adolfo Suárez Madrid–Barajas Airport	Madrid	Spain	MAD	0.022659	0.362690	0.091706	0.123239	0.150073
3482	Hartsfield Jackson Atlanta International Airport	Atlanta	United States	ATL	0.029396	0.357651	0.126460	0.081286	0.148698
191	Lester B. Pearson International Airport	Toronto	Canada	YYZ	0.042527	0.372411	0.085572	0.088808	0.147330
1186	Barcelona International Airport	Barcelona	Spain	BCN	0.010296	0.345412	0.095210	0.129428	0.145087
1633	Zürich Airport	Zurich	Switzerland	ZRH	0.007802	0.364082	0.079731	0.126192	0.144452
497	London Gatwick Airport	London	United Kingdom	LGW	0.010020	0.346927	0.096379	0.114961	0.142072
3295	Newark Liberty International Airport	Newark	United States	EWR	0.017393	0.366008	0.089077	0.094038	0.141629
299	Brussels Airport	Brussels	Belgium	BRU	0.007807	0.349079	0.085572	0.122358	0.141204

Pie chart and Bar chart representation of Top 20 Busiest Airports:





CONCLUSION

Therefore, in this project we have implemented centrality measures and analysed the data of world cities airports a found most influential airports in the world on basis of centrality measures. We have learnt about network analysis and concepts of centrality in real time data applications. And this has helped in improving our skills of programming.

REFERENCES

[1]. SunXiaoqia , VolkerGollnick, SebastianWandelt School of Electronic and Information Engineering, Beihang University, 100083 Beijing, China . Beijing Key Laboratory for Network-based Cooperative ATM, Beijing 100083, China. Beijing Laboratory for General Aviation Technology, Beijing 100083, China

[2]. Dorothy P. Cheung , Mehmet Hadi Gunes , University of Nevada, Reno, Reno, NV, USA 2012 IEEE/ACM International Conference on Advances in Social Networks Analysis and Mining
INSPEC Accession Number: 13287093

[3]. Saleena. P , P. K. Swetha , D. Radha
Department of Computer Science & Engineering, Amrita School of Engineering, Bengaluru2, Amrita Vishwa Vidyapeetham, India *Corresponding author E-mail: d_radha@blr.amrita.edu

[4]. Min Geun SONGa, Gi Tae YEOb a Ph.D., Incheon National University, Korea, E-mail: haoweather@naver.com (First Author) b Professor, Incheon National University, Korea, E-mail:ktyeo@inu.ac.kr(Corresponding Author)