Mini Project 3 – Indian Flights Analysis

Shihabur Rahman Samrat

1. Introduction

This project centers on the network analysis of a dataset containing vital Indian flight information. The dataset encompasses flight numbers, airlines, origins, destinations, schedules, and validity periods. The primary aim is to construct a network illustrating relationships between airports and flights.

Efficient airline scheduling and planning demand a thorough understanding of the air travel network. Identifying major hubs, efficiently allocating resources, and recognizing under-connected airports for strategic expansion is crucial for meeting passenger demands. The project's approach involves creating a network where airports are nodes and flights are edges, followed by the application of network analysis techniques. Metrics like degree centrality, betweenness centrality, and closeness centrality extract valuable insights.

Post-project completion, the focus shifts to identifying significant hubs, exploring route patterns, and revealing potential new routes. Network analysis provides a comprehensive understanding of the Indian air travel landscape, influencing decisions for enhanced efficiency and connectivity.

2. Methodology - Dataset

The dataset for this project was collected from Kaggle. The dataset contains details of Indian flight schedules from 2018-2021. It was collected from data.gov.in. There are two files in the dataset:

Flight_Schedule.csv - contains the directly downloaded file with one column, 'timezone' removed, as it was an exact duplicate of the 'validTo' column.

Flight_Schedule_without_missing.csv - Column 'timezone' has been removed, and rows containing missing values for any of the three columns' airline', 'origin', 'destination' have been removed. It still contains missing values in 'scheduledDepartureTime' and 'scheduledArrivalTime' columns.

For both the files, the columns give the flight#, Origin, Destination, Days of the Week on which the flight operates as per the schedule, Scheduled

Departure and Arrival time and the Start and End dates between which the schedule is valid.

For my project, I used the latter (Flight_Schedule_without_missing.csv), since it already had some preprocessing done.

The main motivation to select this dataset was the size of the dataset, which was not too large. Since the project needed to be done on Google Colab, I needed to keep in mind the computational limitations of Colab.

Firstly, after loading the dataset as a dataframe, I checked the number of columns as well as the number of values in each column.

```
RangeIndex: 31680 entries, 0 to 31679
Data columns (total 9 columns):
 #   Column                 Non-Null Count  Dtype
---  ------                 --------------  -----
 0   flightNumber           31680 non-null  object
 1   airline                31680 non-null  object
 2   origin                 31680 non-null  object
 3   destination            31680 non-null  object
 4   dayOfWeek              31680 non-null  object
 5   scheduledDepartureTime 21923 non-null  object
 6   scheduledArrivalTime   21733 non-null  object
 7   validFrom              31680 non-null  object
 8   validTo                31680 non-null  object
```

Figure 1: Dataset columns

Besides this, I also printed out the number of missing values (if any) in each column.

```
Missing Values:
flightNumber                0
airline                     0
origin                      0
destination                 0
dayOfWeek                   0
scheduledDepartureTime    9757
scheduledArrivalTime      9947
validFrom                   0
validTo                     0
```

Figure 2: Missing values in dataset

As we can see above, there are quite a few missing values in the column "scheduledDepartureTime" and "scheduledArrivalTime". However, these columns are not considered when we do our network analysis, which is why this is not a problem.

3. Methodology – Approach

The library used for doing the network analysis was **networkx.** I used this library to create a directed graph, where the airports represented the nodes, and the edges represented the flights between them. I also added the flight number and airlines as attributes to the edges, however, they did not play a significant role in this project. They may be used for further analysis in the future.

As mentioned above, I needed to make sure that the network is not too complex for Google Colab, which is why I checked the number of nodes and edges right after creating the network.

Number of airports (nodes): 74

Number of routes (edges): 952

As we can see, the number of nodes and edges are small enough to be handled by Google Colab while being large enough to do some analysis.

Before calculating the network metrics, I visualized the network using spring layout and tried to identify the communities in the network.

In the context of a flight network, communities identified through community detection algorithms reveal clusters of airports that exhibit denser connections among themselves compared

to those outside the community. These communities offer valuable insights into the underlying structure and organization of the air travel network. Possible interpretations include geographical proximity, where communities may signify airports within the same region; airline alliances, as airports within a community could be hubs or destinations associated with specific airline alliances; similar traffic patterns, indicating airports serving comparable flight types or connecting to common destinations. These interpretations collectively contribute to a thorough understanding of the complexities within the air travel network.

After identifying the communities, I calculated several network metrics that can provide valuable insights. Let's talk about each of these metrics and understand how they may be helpful in the context of flights.

A. Degree Centrality

Degree centrality dictates the number of flights (edges) connecting to a particular airport (node). Airports with a higher degree centrality are more connected, indicating more direct flights to and from other airports. They act as major hubs in the network.

B. Betweenness Centrality

Betweenness centrality is the fraction of shortest paths between all pairs of airports that pass through a particular airport. Airports with higher betweenness centrality act as critical connectors in the network. They play a crucial role in facilitating connectivity between different regions, even if they don't have the highest number of direct flights.

C. Closeness Centrality

Closeness centrality is the reciprocal of the average shortest path length from a particular airport to all other airports. Airports with higher closeness centrality are more accessible to other airports. They can be considered as airports that are well-connected and reachable within a shorter number of flights.

D. Eigenvector Centrality

Eigenvector centrality in the context of air travel assigns scores to airports based on both their connections and the centrality of connected airports, emphasizing not just the quantity but the importance of connections. Airports with high eigenvector centrality are influential hubs connected to other pivotal airports, playing a critical role in efficiently connecting different regions in the air travel network. This metric serves as a valuable tool for identifying key airports that facilitate seamless travel and contribute significantly to the overall network connectivity.

E. Clustering Coefficient

The clustering coefficient in the context of a flight network signifies the tendency of airports within a local neighborhood to form densely connected clusters. A higher clustering coefficient suggests well-integrated air travel routes within specific regions, showcasing cohesive connections

between airports. Conversely, a lower coefficient may indicate a more dispersed network, reflecting a less centralized or organized air travel structure within certain areas.

F. In-degree and Out-degree

The in-degree of an airport is the number of incoming flights (edges) it has. In other words, it represents the number of flights arriving at that airport. Airports with high in-degrees are destinations for many flights. In the context of air travel, airports with high in-degrees are likely to be major hubs or popular destinations. The out-degree of an airport is the number of outgoing flights (edges) it has. It represents the number of flights departing from that airport. Airports with high out-degrees have many outgoing flights, indicating that they serve as departure points for many destinations. These airports might play a significant role in connecting different regions.

4. Results

Now that we know the methods and approaches that I used, let's delve deeper into the results I got. Firstly, let us visualize the entire network. It may be difficult to see each node in this paper, but in the Google Colab file, it is clearer.

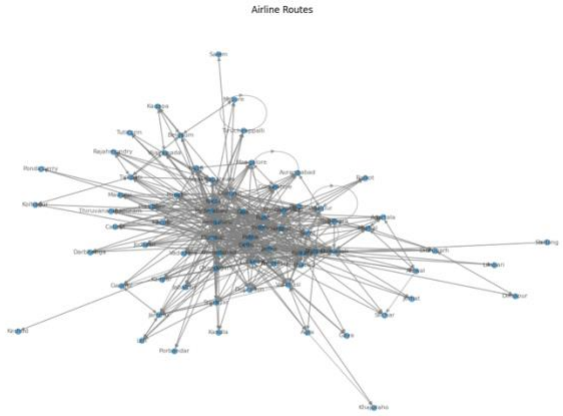

Figure 3: Full network visualization

Next, let us see the communities that were detected. Overall, I detected three communities, and their graphs can be seen below.

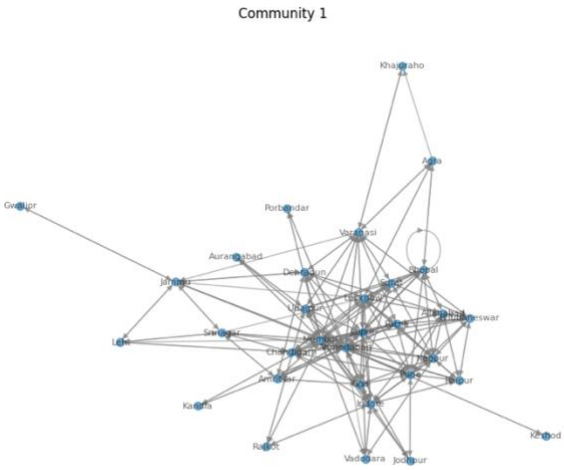

Figure 4: Community 1 visualization

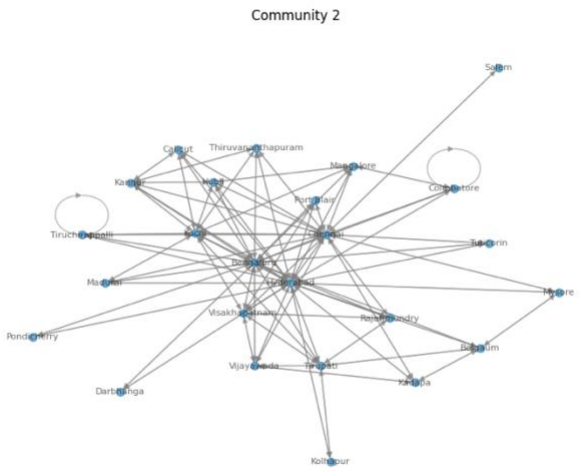

Figure 5: Community 2 visualization

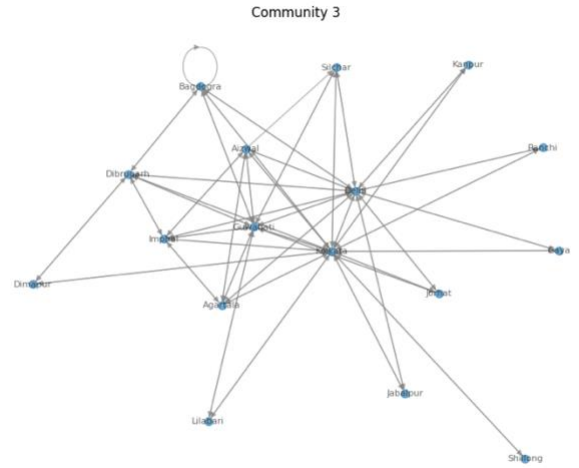

Figure 6: Community 3 visualization

Now, let's look at the results of each of the metrics and understand what they mean.

Degree Centrality

```
Top 5 airports by degree centrality:
1. Delhi: 1.52
2. Bengaluru: 1.48
3. Hyderabad: 1.47
4. Mumbai: 1.38
5. Kolkata: 1.22

Bottom 5 airports by degree centrality:
1. Keshod: 0.03
2. Salem: 0.03
3. Shillong: 0.03
4. Khajuraho: 0.04
5. Dimapur: 0.05
```

Figure 7: Degree centrality results

Delhi, Bengaluru, Hyderabad, and Mumbai are major hubs with the highest degree centrality.

Keshod, Salem, Shillong, Khajuraho, and Dimapur have lower connectivity.

Betweenness Centrality

```
Top 5 airports by betweenness centrality:
1. Delhi: 0.16
2. Hyderabad: 0.14
3. Kolkata: 0.13
4. Bengaluru: 0.13
5. Mumbai: 0.11

Bottom 5 airports by betweenness centrality:
1. Ranchi: 0.00
2. Port Blair: 0.00
3. Aurangabad: 0.00
4. Rajkot: 0.00
5. Jodhpur: 0.00
```

Figure 8: Betweenness centrality results

Delhi has the highest betweenness centrality, indicating its critical role in connecting different parts of the network.

Ranchi, Port Blair, Aurangabad, Rajkot, and Jodhpur have lower betweenness centrality.

Closeness Centrality

```
Top 5 airports by closeness centrality:
1. Delhi: 0.80
2. Bengaluru: 0.79
3. Hyderabad: 0.78
4. Mumbai: 0.76
5. Kolkata: 0.72

Bottom 5 airports by closeness centrality:
1. Khajuraho: 0.37
2. Keshod: 0.40
3. Salem: 0.42
4. Shillong: 0.42
5. Dimapur: 0.42
```

Figure 9: Closeness centrality results

Delhi has the highest closeness centrality, indicating its proximity to other airports in the network.

Khajuraho, Keshod, Salem, Shillong, and Dimapur are relatively less accessible.

Eigenvector Centrality

```
Top 5 airports by eigenvector centrality:
1. Bengaluru: 0.27
2. Delhi: 0.27
3. Hyderabad: 0.27
4. Mumbai: 0.26
5. Chennai: 0.24

Bottom 5 airports by eigenvector centrality:
1. Khajuraho: 0.01
2. Keshod: 0.01
3. Shillong: 0.01
4. Salem: 0.01
5. Dimapur: 0.01
```

Figure 10: Eigenvector centrality results

Bengaluru, Delhi, Hyderabad, Mumbai, and Chennai are influential airports based on their connections and the connections of their connections.

Khajuraho, Keshod, Shillong, Salem, and Dimapur have lower influence.

Clustering Coefficient

```
Top 5 airports by clustering coefficient:
1. Ranchi: 1.00
2. Port Blair: 1.00
3. Aurangabad: 1.00
4. Rajkot: 1.00
5. Jodhpur: 1.00

Bottom 5 airports by clustering coefficient:
1. Keshod: 0.00
2. Salem: 0.00
3. Shillong: 0.00
4. Delhi: 0.24
5. Hyderabad: 0.25
```

Figure 11: Clustering coefficient results

Ranchi, Port Blair, Aurangabad, Rajkot, and Jodhpur have a perfect clustering coefficient, indicating they form closed clusters.

Keshod, Salem, Shillong, Delhi, and Hyderabad have lower clustering coefficients, suggesting they are less clustered.

In-Degree and Out-Degree

```
Top 5 airports by in-degrees:
1. Delhi: 55
2. Bengaluru: 54
3. Hyderabad: 53
4. Mumbai: 50
5. Kolkata: 45

Bottom 5 airports by in-degrees:
1. Keshod: 1
2. Salem: 1
3. Shillong: 1
4. Khajuraho: 2
5. Dimapur: 2
```

Figure 12: In-degree results

```
Top 5 airports with the out-Degrees:
1. Delhi: 56
2. Bengaluru: 54
3. Hyderabad: 54
4. Mumbai: 51
5. Kolkata: 44

Bottom 5 airports by out-degrees:
1. Khajuraho: 1
2. Keshod: 1
3. Salem: 1
4. Shillong: 1
5. Dimapur: 2
```

Figure 13: Out-degree results

Delhi, Bengaluru, Hyderabad, Mumbai, and Kolkata have the highest in-degrees and out-degrees, indicating their significance in both incoming and outgoing flights.

Khajuraho, Keshod, Salem, Shillong, and Dimapur have lower in-degrees and out-degrees.

Comments

Connectivity Hubs: Delhi, Bengaluru, Hyderabad, and Mumbai emerge as major connectivity hubs with high centrality in various measures.

Under-Connected Airports: Airports like Keshod, Salem, Shillong, Khajuraho, and Dimapur appear to be relatively under-connected and less central in the network.

Critical Connectors: Delhi plays a crucial role as it tops in degree, betweenness, and closeness centrality.

Clustering and Community Structure: Some airports form tightly knit clusters (Ranchi, Port Blair, Aurangabad, Rajkot, Jodhpur), while others have lower clustering coefficients, indicating a potential lack of community structure.

5. Conclusion

In conclusion, this project revolves around conducting a network analysis of a comprehensive Indian flight dataset to construct a visual representation of relationships between airports and flights.

The main challenge was to find a dataset that is appropriate to be used on Google Colab. This is because majority of the datasets are extremely large, and Google Colab cannot handle this many nodes/edges in a suitable amount of time. Besides this, visualizing networks with many edges/nodes is also a tedious task.

However, despite the challenges, I feel my goal with this mini project has been accomplished. Upon network analysis, I was able to identify the airports that were major hubs, critical connectors, and also under-connected airports. In terms of future work, different attributes can be analyzed, such as flight number and airlines, which can help us observe even more interesting aspects of air travel.