Live Air Traffic Network

Nicola Gastaldello (IDNumber: 1206748)

January 16, 2019

In this second paper, I continue the work on Live Air Traffic Network. In the first work, I analyzed the structure of the network and its main property like degree distribution, assortativity, clustering coefficient and robustness. Now, I have done analysis of the communities.

1 Introduction

While in the first work I analyzed three datasets and compared them, for this work I decided to use only one dataset, because some algorithms have high computational time and they are memory intensive. The algorithms used are PageRank and HITS, to understand the nodes ranking, modularity, PageRank-Nibble approach and spectral approach to detect the communities, AUC to evaluate the performance in link prediction and finally I simulated SEIR epidemics.

2 Ranking algorithms

To rank nodes I used two algorithms: PageRank and HITS for hubs and authorities. The approaches used were the power iteration method and the solutions of the linear system.

2.1 PageRank

PageRank is an algorithm that works counting the number and the quality of links to a node to determine its importance. Applying this to my air traffic network I noticed that the most important is London International Airport (YXU) which is one of the most significant airport in the world, but not the busiest one. I also noticed that in the top ten by PageRank there are not only the most important airports, because there could be dead ends. Moreover, some small airports are connected to large airports and the random walk tends to reach it often, so the score is high.

Node	**Rank**
YXU	0.0097733
HHN	0.0089738
MSN	0.0085304
PIN	0.0071875
DYR	0.0070174
SVC	0.0066155
HOX	0.0064043
ISP	0.0058855
KID	0.0055826
PEK	0.005407

Table 1: The first ten airports rank by the PageRank algorithm.

2.2 HITS

According to HITS algorithm, the network is divided in two classes: authorities and hubs. The first are airports with a lot of arrivals and the second have many departures. As expected, the bigger hub is Hartsfield-Jackson Atlanta International Airport and the bigger authorities is Beijing Capital Airport, two of the most significant airports. Also in this case, there are some airports not very important.

Hubs	**Score**	**Autorithies**	**Score**
ATL	6.5052e-19	PEK	3.4694e-18
ABA	5.421e-19	OMA	1.7347e-18
KKE	5.421e-19	MNZ	8.6736e-19
NWI	5.421e-19	LAX	5.421e-19
SHF	5.421e-19	DLA	4.3368e-19
OSS	4.3368e-19	HAJ	4.3368e-19
TBH	4.3368e-19	LBL	4.3368e-19
PSA	2.1684e-19	AGH	2.7105e-19
BKO	1.0842e-19	AEP	2.7105e-19
KSU	1.0842e-19	YQI	2.1684e-19

Table 2: The first ten airports rank by the HITS algorithm for hubs and authorities.

3 Community detection algorithms

Network are composed by communities, which are generically defined as sets of nodes that are more connected among themselves than with the rest of the network. There are several algorithms to make community detection and they have been used to discover potential community structures and to group airports. I chose three of them: modularity, PageRank-Nibble and spectral clustering.

3.1 Modularity

To implement modularity I used python, because I found a good library that guaranteed good output in optimal time. This library uses the 'Louvain method', an iterative, multistage modularity maximization approach that has been shown to detect communities with high modularity. The clustering obtained is good, according to the theory, because the modularity coefficient is $0,3 < Q = 0.54476 < 0,7$. Moreover, the results provide by the algorithm are good, because the communities are distributed around geographically.

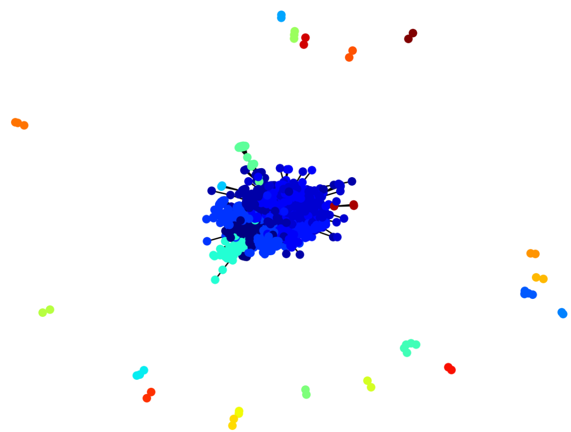

Figure 1: Community detection with modularity optimization

3.2 PageRank-Nibble

The PageRank-Nibble approach is another way to detect community. It is based on the PageRank algorithm, because it discovers clusters based on the node with highest connection. For this algorithm I tried to approach using only the connected component of the air traffic network with bad results and so I used the entire network. The size of the two communities found by the PageRank-Nibble algorithm are 1262 and 2970. The minimum conductance value is 0.01818.

3.3 Spectral clustering

The spectral approach is an algorithm to detect community and it divides the network in two groups. The size of the two communities are 645 and 3587. The minimum conductance value is 0.48619.

4 Link prediction

In recent years, the number of air passengers is increasing, and according to this trend, it is assumed that the demand for the air routes will increase. Link prediction can be very useful for air transportation network optimization, for example through adding one or more flight routes.

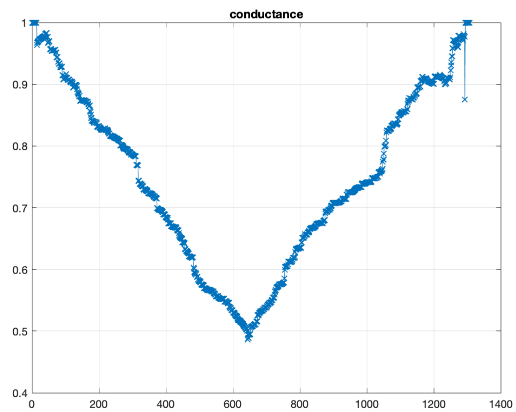

Figure 2: Conductance plot with spectral clustering approach

The prediction measures are based on the similarity of the node pairs (two airports) calculated only from the network structure. In this paper I used 8 measures which are reported in the Table 3.

	Test 1	**Test 2**	**Test 3**	**Test 4**	**Test 5**	**AUC average**
Common Neighbours	0.71423	0.70828	0.71337	0.71275	0.70861	0.71144
Adamic Adar	0.71351	0.70831	0.71155	0.71205	0.70789	0.71066
Resource Allocation	0.71422	0.70828	0.71338	0.71277	0.70859	0.71145
Local Path	0.76881	0.76304	0.78028	0.76881	0.76451	0.76909
Kats	0.77748	0.76954	0.78369	0.77763	0.77534	0.77673
Superposed Random Walk	0.80329	0.79416	0.80482	0.79785	0.79389	0.79880
Local Random Walk	0.80326	0.79415	0.80475	0.79776	0.79384	0.79875
Random Walk with Restart	0.80308	0.79525	0.80665	0.80073	0.79705	0.80055

Table 3: Comparison of AUC performance of link prediction methods.

To evaluate the AUC performance, I made five tests, splitting random the dataset each time, and then I compute AUC mean. The highest value I found is $AUC = 0.80055$, which is the result given by Random Walk with restart.

5 Epidimics simulation

For epidemics simulation I used the SEIR model, which expects 4 different states for each node: Susceptible, Exposed, Infected, Recovered. I assigned a state to each node: 0 for Susceptible and -1 for Recovered. For the Exposed and the Infected state I built a vector with 8 cells, one for each different state of disease and each by a parameter (infectiousness), which determines the probability that a node at that state infects a node in state S. I randomly selected 10 nodes that are infected at the beginning of the epidemic. We reach a disease-free equilibrium after few time units, meaning that the basic reproductive ratio is smaller than 1. Therefore, the spreading of the disease is very fast.

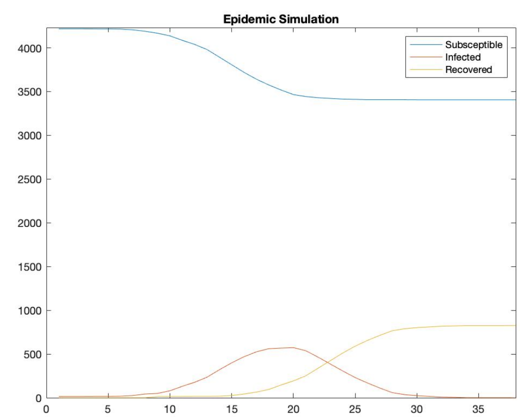

Figure 3: Result of epidemics simulation

6 Conclusion

In this second paper are shown important analysis of the communities based on the live air traffic network, which is one of the most important network today, because people who use the plane are increasing significantly. The results obtained are in accord with the theory and with what I expected.