Forbes Global 2000 Opening Stock Market Values Correlation Network Analysis

Khoa Nguyen (H272580), Rostislav Duda (H268465)
Group: crc32b
Tampere University
Tampere, Finland
khoa.nguyen@tuni.fi, rostislav.duda@tuni.fi

I. Introduction

Stock market is a stochastic dynamic system that has fascinated economists, complexity scientists, and common folk wishing to make financial gains for a very long time. One might consider it to be one of the most, if not the most, complex system created by mankind. It is only natural for one to wish to tame this complexity or at least attempt to comprehend it, and so the goal of this project is to study the interconnectedness of various financial assets. Specifically, it might prove to be quite interesting to study the correlation between stock value time series of top-performing public companies. Analysis of such a graph could prove to be a highly beneficial endeavor in the context of financial forecasting, since if we spot a reasonably stable correlation trend between some assets, we could make smart investment choices related to one of assets based on what we currently know about another. The structure of this report is as follow: section II reports the process of gathering, processing the data, constructing the network from the data. Section III contains three different parts: in III-A, we compare the statistics of our network with a random network and a scale-free network of the same size; then three methods for discovering network communities are reported in III-B;

II. Data Collection

First, the data of the 2000 the world's largest public companies (as of 2020) has been gathered from Forbes 2000 website[1]. The resulting data contained six columns: company name, country, sales, profits, assets, and market value, as shown in Figure 1. We were mostly interested in the names of the companies, as our main goal was to investigate only the network of opening market values over time. Nonetheless, other data such as country, sales, profits, etc. could be used later for more advanced analysis.

Company	Country	Sales	Profits	Assets	Market Value
ICBC	China	177230	45284	4322528	242283
China Construction Bank	China	162147	38915	3822048	203818
JPMorgan Chase	United States	142927	29954	3139431	291737
Berkshire Hathaway	United States	254616	81417	817729	455444
Agricultural Bank of China	China	148692	30912	3697451	147174
Saudi Arabian Oil Company (Saudi Aramco)	Saudi Arabia	329762	88205	398296	1684765
Ping An Insurance Group	China	154952	18751	1218615	187215
Bank of America	United States	112065	24129	2619954	208646
Apple	United States	267712	57215	320400	1285516
Bank of China	China	135353	27216	3387044	112774
AT&T	United States	179217	14417	545354	218558
Toyota Motor	Japan	280529	22682	495064	173300
Alphabet	United States	166310	34522	273403	919284
ExxonMobil	United States	255995	14340	362597	196588
Microsoft	United States	138604	46266	285449	1359028
Samsung Electronics	South Korea	197568	18441	304868	278742
Wells Fargo	United States	98866	14342	1981349	118797
Citigroup	United States	104423	17091	2219770	101102
Walmart	United States	523964	14881	236495	344436
Verizon Communications	United States	131350	18389	294500	237728
Royal Dutch Shell	Netherlands	311562	9947.8	393961	126489
Amazon	United States	296274	10562	221238	1233351

Fig. 1. Data scraped from Forbes 2000 website.

After the names of the companies have been gathered, we needed to get their tickers on the stock market. Tickers are the continuously updated records of the prices of company securities, which we use to fetch the opening market values. This has been done by making a search query of form `yahoo finance <company name>` for each of the 2000 companies in an automated fashion, getting the first resulting link, and extracting the ticker from the link. Examples of some tickers are shown in Figure 2.

ICBC	1398.hk
China Construction Bank	0939.hk
JPMorgan Chase	JPM
Berkshire Hathaway	BRK-A
Agricultural Bank of China	ACGBY
Saudi Arabian Oil Company (Saudi Aramco)	2222.SR
Ping An Insurance Group	2318.hk
Bank of America	BAC
Apple	AAPL
Bank of China	3988.hk
AT&T	T
Toyota Motor	TM
Alphabet	GOOG
ExxonMobil	XOM
Microsoft	MSFT
Samsung Electronics	005930.KS
Wells Fargo	WFC
Citigroup	C
Walmart	WMT
Verizon Communications	VZ
Royal Dutch Shell	RDS-A
Amazon	AMZN

Fig. 2. Example tickers.

Some of the tickers gathered in the scraping process were found to be invalid, as their values did not correspond to the underlying security (e.g. entries like "history", "profile" etc), and as such have been removed from our set of tickers. It could be the case that the corresponding companies went bankrupt, or have stopped listing their data on Yahoo. Figure 3 illustrates some examples of the companies with improper tickers. After removing them from the dataset, we ended up with 1969 tickers.

The next step was to download the opening stock market values of every company for the period of 10 years with 1 day interval. We have used the API called `yfinance`[2] for

[1] https://www.forbes.com/global2000

[2] https://github.com/ranaroussi/yfinance

CIT Group	CIT
Bank of Qingdao	news
Bohai Leasing	000415.SZ
Hiroshima Bank	profile
WEG	WEGE3.SA
Rajesh Exports	history
Safaricom	safaricom-vodacom-finalise-m-pesa-111210131.html
Johnson Matthey	JMAT.I
INTL FCStone	1

Fig. 3. Example bad tickers (marked in red).

downloading the data from Yahoo Finance[3]. There were 14 companies for which stock data could not be found on Yahoo, resulting in a dataset of 1955 companies with opening stock values in the period from March 4th, 2011 to March 4th, 2021, as illustrated in Figure 4.

Date	1398.hk	0939.hk	JPM	BRK-A	ACGBY	2222.SR	2318.hk
2011-03-04	6.1300001144409	6.960000038147	45.979999542236	129900	0	0	41.150001525879
2011-03-07	6.1199998855591	6.9699997901917	45.619998931885	128500	0	0	41.25
2011-03-08	6.1799998283386	7	45.380001068115	128035	0	0	41.525001525879
2011-03-09	6.25	7.1399998664856	46.330001831055	129245	0	0	42.5
2011-03-10	6.2800002098084	7.2199997901917	46.080001831055	128000	0	0	42.849998474121
2011-03-11	6.2399997711182	7.170000076294	45.279998779297	126400	0	0	41.849998474121
2011-03-14	6.0999999046326	7.0999999046326	45.419998168945	126616	0	0	40.849998474121
2011-03-15	6.1500000953674	7.0599999427795	44.549999237061	124719	0	0	39
2011-03-16	6.1199998855591	7.039999961853	44.639999389649	124200	0	0	38.95000076294
2011-03-17	6.0700001716614	6.9000000953674	44.459999084473	123075	0	0	37.400001525879
2011-03-18	6.0100002288818	6.9400000572205	45.240001678467	125401	0	0	37.174999237061
2011-03-21	5.9499998092651	7	46.279998779297	126245	0	0	37.150001525879
2011-03-22	6.0799999237061	7.0300002098084	45.659999847412	127875	0	0	37.70000076294
2011-03-23	6.1199998855591	7.0999999046326	45.290000915527	127200	0	0	38.20000076294
2011-03-24	6.210000038147	7.1500000953674	45.70000076294	127520	0	0	38.974998474121
2011-03-25	6.3800001144409	7.3000001907349	45.70000076294	127950	0	0	38.95000076294
2011-03-28	6.4000000953674	7.2699999809265	46.099998474121	127346	0	0	38.724998474121
2011-03-29	6.3400001525879	7.1500000953674	45.830001831055	126200	0	0	38.07500076294
2011-03-30	6.3800001144409	7.210000038147	46.529998779297	127222	0	0	37.650001525879
2011-03-31	6.4899997711182	7.3200001716614	46.049999237061	125956	0	0	39.049999237061

Fig. 4. Stock data of 1955 companies from March 4th, 2011 to March 4th, 2021.

Finally, the pairwise Pearson correlation coefficients for all companies have been calculated as follows:

$$r_{xy} = \frac{\sum_{t=1}^{N}(x(t) - \mathbb{E}[x])(y(t) - \mathbb{E}[y])}{\sqrt{\sum_{t=1}^{N}(x(t) - \mathbb{E}[x])^2}\sqrt{\sum_{t=1}^{N}(y(t) - \mathbb{E}[y])^2}}. \quad (1)$$

Pearson correlation coefficient yields a value $r_{xy} \in [-1, 1]$, with –1 corresponding to complete anticorrelation and 1 to complete correlation. A link between two nodes was constructed if the correlation coefficient was greater than 0.8, so only highly positively correlated stock pairs have been linked. The attribute "country" where a company originally locates is also added to each node. Figure 5 shows the network in Yifan Hu layout. The network has 1955 nodes, 139561 edges, average degree of 142.773 and network density of 0.073.

III. Network Analysis

A. Comparison to ER and BA Network Models

As the first task in our analysis, we have generated synthetic networks using Erdős–Rényi (ER), and Barabási–Albert construction with similar numbers of nodes and edges as our original stock network. For ER network, this has been accomplished by a brute-force generation of $G(N, p)$ random networks with parameters set to $N = 1955$, $p = 0.073067$ until obtaining a network with a number of edges $L = 139561$. As for BA network, we proceeded by generating networks $BA(N, m)$ with $N = 1955$, $m = \lfloor \frac{1}{2}\langle k \rangle \rfloor = 71$, where m is

[3]https://finance.yahoo.com/

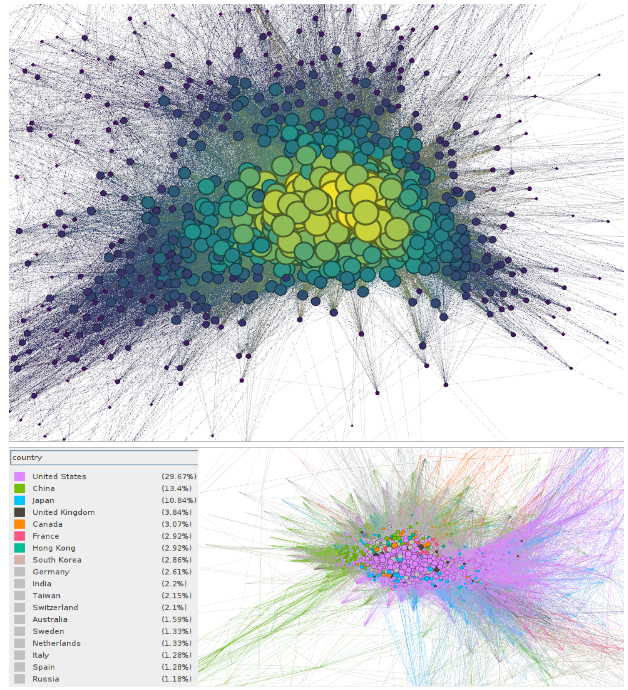

Fig. 5. The stock opening value correlation network. Top: illustration of the network with size and color of nodes based on the ranking of degrees. Bottom: illustration of the network with size of nodes are based on degree ranking, and colors are based on the countries where the companies are from.

the number of edges to attach from a new node to existing nodes. After each iteration, m has been increased by 1 until the number of edges in the generated network was such that $|L_{BA} - L| = |L_{BA} - 139561| < 500$. Figures below visualize the two synthetic networks:

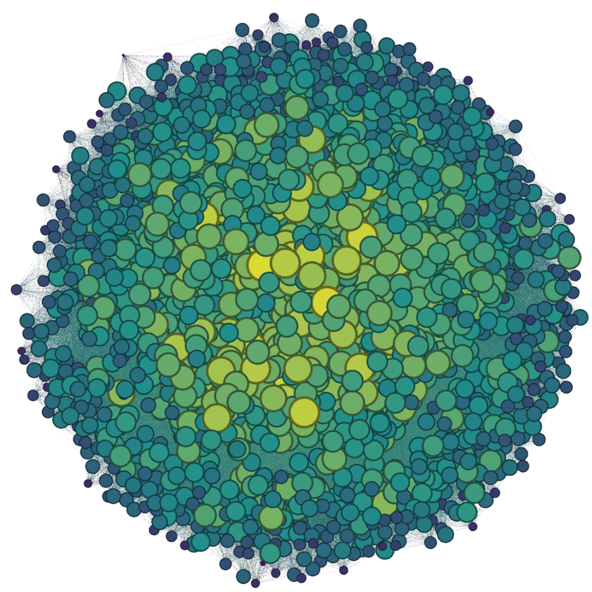

Fig. 6. Erdős–Rényi (ER) random network model.

The following table summarizes some of the descriptive statistics of the three models:

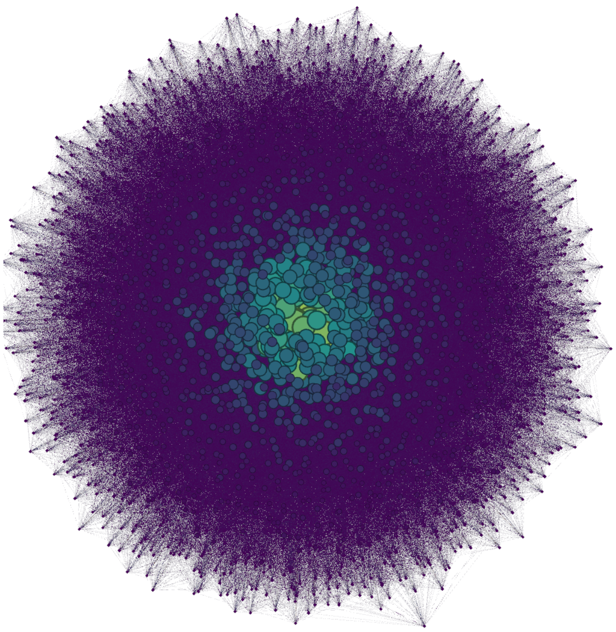

Fig. 7. The Barabási–Albert (BA) preferential attachment network.

	N	$\ln N$	L	k_{min}	k_{max}	$\langle k\rangle$	$\langle d\rangle$	$\langle C\rangle$	ρ
Stock	1955	7.58	139561	0	570	142.77	3.86	0.57	0.073
ER	1955	7.58	139561	103	182	142.77	1.93	0.073	0.073
BA	1955	7.58	139194	74	649	142.40	1.93	0.14	0.073

TABLE I
SUMMARY STATISTICS OF THE SYNTHETIC NETWORKS IN COMPARISON TO OUR OWN CRAWLED NETWORK. HERE $\langle d\rangle$ IS THE AVERAGE PATH LENGTH, $\langle C\rangle$ IS THE AVERAGE CLUSTERING COEFFICIENT AND ρ IS THE NETWORK DENSITY.

We note that overall the numbers of nodes, edges, average degree and network density are very similar in all models, as is to be expected by construction. As a sidenote, the average degree of our network is quite a bit larger than most real-life networks presented on the course, with the actor network coming close at $\langle k\rangle = 83.71$. This tells us that economy as a complex system is a highly interconnected one. This hypothesis is further confirmed by the average clustering coefficient of our network $\langle C\rangle$, that is about 4 times larger than that of the corresponding Barabási–Albert synthetic model. Since scale-free networks with lower clustering exhibit an ultra-small world property already, we can likely say the same about our network. Figure below provides a comparison between the distributions of clustering coefficients in the three networks:

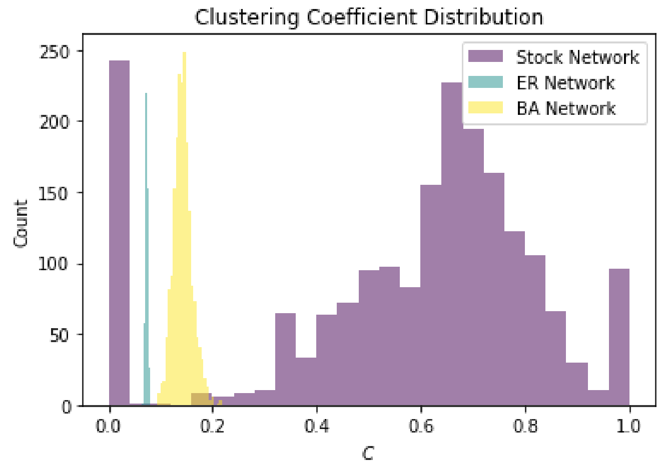

Fig. 8. Clustering coefficient comparison.

We can observe that our stock network has a wider clustering spread centered in the region of larger clustering values compared to more localized and smaller value clustering coefficients of the synthetic networks. A large jump at 0 clustering seen in the stock network clustering distribution is due to a sizeable fraction of the nodes being completely isolated from the rest of the graph.

Next, we look at the degree distributions of the three networks:

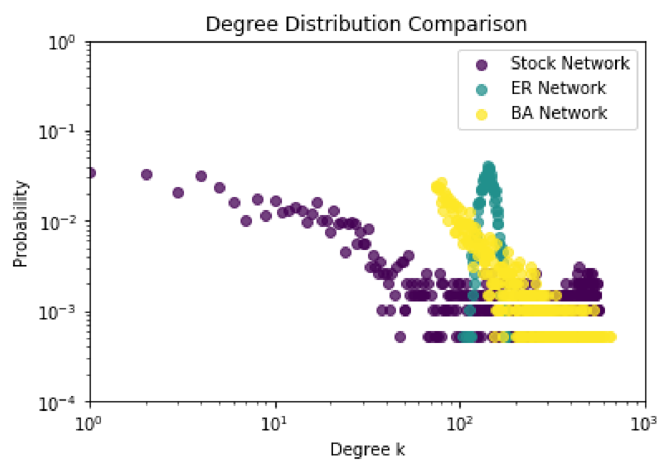

Fig. 9. Degree distribution comparison.

We observe that the Poisson distribution characterising the ER model has been heavily skewed to the right towards higher degree values, which is likely due to the network converging to the connected regime, since $\langle k\rangle > \ln N$. In this mode, most nodes collapse into a giant component and are likely to have high degree values, and we can note that interconnectedness also when looking at Figure 6. Looking at the stock network, it seems to follow the power law scaling similar to the BA network, although the slope of the curve seems to be slightly smaller than $\gamma = 3$ in comparison, but still in the "real-world" network interval.

Finally, we examine centrality for each of the networks. Specifically, we employ five different measures: degree centrality, eigenvalue centrality, PageRank centrality, closeness centrality and betweenness centrality. We focus on comparing the largest centrality values found in each network, as well as the average centrality values.

$\max(c_i)$	Degree	Eigenvalue	PageRank	Closeness	Betweenness
Stock	0.291709	0.056366	0.001268	0.342031	**0.030032**
ER	0.093142	0.028777	0.000631	0.524423	0.000762
BA	**0.332139**	**0.085409**	**0.002054**	**0.599570**	0.006989

TABLE II
COMPARISON OF MAXIMUM CENTRALITIES FOR EACH OF THE THREE NETWORKS. WINNERS IN EACH CATEGORY ARE MARKED IN BOLD.

We can see that overall, Barabási–Albert synthetic model has the nodes with largest maximum centrality for all centrality types except for betweenness, where the stock network outperforms its competitors. Proceeding with average centralities for each graph, we get the following results:

$\langle c_i \rangle$	Degree	Eigenvalue	PageRank	Closeness	Betweenness
Stock	**0.073067**	0.012882	**0.000512**	0.219011	**0.001128**
ER	**0.073067**	**0.022544**	**0.000512**	**0.518958**	0.000475
BA	0.072875	0.018955	**0.000512**	0.518930	0.000475

TABLE III
COMPARISON OF AVERAGE CENTRALITIES FOR EACH OF THE THREE NETWORKS. WINNERS IN EACH CATEGORY ARE MARKED IN BOLD.

Mainly, we can see that on average, degree and PageRank centrality measures for all graphs are quite similar. Our stock network in this comparison has the smallest eigenvalue and closeness centrality, and the largest betweenness centrality. The large betweenness centrality of the stock network could potentially be explained by the fact that the average path length of the network is twice that of the synthetic models, so on average there are more nodes on the path from one to another, leading to more nodes in between each node. Additionally, table below records the top 10 stocks in each centrality measure:

	Degree	Eigenvalue	PageRank	Closeness	Betweenness
1	AVY	TXN	BEN	LMT	BMO
2	TXN	HD	PTR	NOC	RDS-A
3	APH	AVY	FB	HLT	TOT
4	AVGO	SYK	APH	ALL	MT
5	HD	IHI	GL	BDX	CVX
6	FB	ICE	STZ	APH	2884.TW
7	SYK	V	AVY	AVY	KBC.BR
8	UNH	TFX	NOC	GL	BCE
9	V	APH	LMT	PKG	SMFG
10	TFX	AVGO	ALL	HON	DNZOY

TABLE IV
FORBES GLOBAL 2000 STOCK CENTRALITY RANKINGS.

B. Community Discovery

Communities help us understand the underlying wiring diagram in a network, as they reveal who connects to whom. A community is defined as a locally dense connected subgraph in a network [5]. In this task, we have used three different communities discovery algorithms, namely Louvain[4], Infomap[5] and Demon[6].

1) Louvain: The Louvain algorithm [2] aims to optimize the modularity of the network. According to [5], the modularity M is defined as

$$M = \sum_{c=1}^{n_c} \left[\frac{L_c}{L} - \left(\frac{k_c}{2L} \right)^2 \right] \tag{2}$$

where where L_c is the total number of links within a community C_c, k_c is the total degree of the nodes in this community, and n_c is the total number of communities. Essentially, a higher modularity implies better community partition [5].

The Louvain algorithms iteratively executes two steps:

- Step 1: first, each node in the graph is assigned to a different community. Then, each node is placed in the communities of its neighbors to find the largest modularity gain. The node i will then move to the community with the largest gain, otherwise, if the gain is 0, it stays still. All nodes go through this process until no further improvements can be achieved.
- Step 2: a new network is constructed with communities discovered in step 1. Nodes that are in the same community will be merged into a single node. Step 1 and step 2 together form a pass of the algorithm.

The algorithm repeats the passes until more changes found and maximum modularity is attained.

The result when running Louvain on the stock network is shown in Figure 10 (top). We can see when comparing the top and bottom part of Figure 10 that there are similarities between the Louvain partitions and the network partitioned by countries. There are 175 communities found by Louvain, compared to 63 countries in total.

2) Infomap: Infomap [3] discovers communities in a network by optimizing an entropy-based measure called the map equation, denoted L and is defined as

$$L = q_{\curvearrowright} H(\mathcal{Q}) + \sum_{i=1}^{m} p_{\circlearrowright}^{i} H\left(\mathcal{P}^{i}\right) \tag{3}$$

where $q_{\curvearrowright}$ is the probability that the random walker switches communities, $H(.)$ is the entropy function defined by Shannon [1], which essentially calculates the amount of randomness of a random variable (the greater the randomness, the higher the entropy); $p_{\circlearrowright}^{i}$ defines the weight for the entropy of movements within community i. Hence, the first term of this equation, i.e. $q_{\curvearrowright} H(\mathcal{Q})$, gives the average number of bits needed to for a node to move between communities, and the second term, i.e. $\sum_{i=1}^{m} p_{\circlearrowright}^{i} H\left(\mathcal{P}^{i}\right)$, gives the average number of bits needed for movement within a community. By minimizing L, Infomap tries to compress the movement of a

[4]https://github.com/taynaud/python-louvain
[5]https://www.mapequation.org/
[6]https://github.com/GiulioRossetti/demon

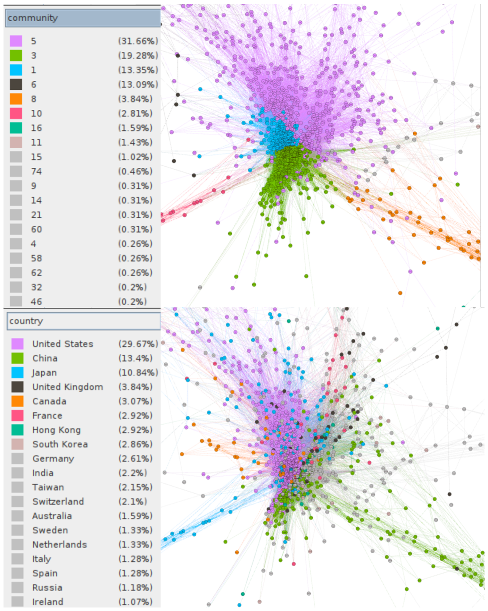

Fig. 10. Communities discovery using Louvain for the stock network.

random walker on a network, such that the random walker's trajectory would take the smallest number of bits possible.

Running Infomap on our stock network reveals 20 communities. The number of nodes are also reduced to 1808 nodes after the algorithm, versus 1955 nodes in the original network. There is one giant community, which consists of 93.58% of all nodes; other communities are tiny and separated, as shown in Figure 11. This result is significantly different than communities partitioned by Louvain and by countries, and does not provide as much information as one community takes almost all the nodes in the network.

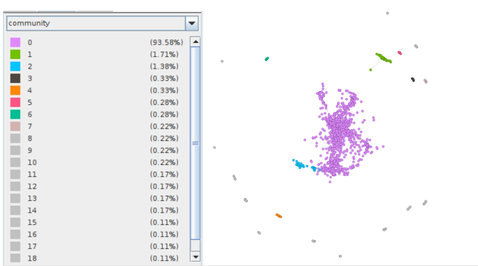

Fig. 11. Communities discovery using Infomap for the stock network.

3) Demon: Demon [4] is an another community discovery algorithm that takes a different approach from Infomap and Louvain: instead of imposing a top-down global view of a network, Demon approaches the problem with a local-first method.

We applied the Demon implementation from[7] with the *epsilon* parameter of 0.25, and the min community size is 20. There is a noticeable difference of the output data structure from Demon compared to that of Infomap and Louvain: for Demon, one node can belong to multiple communities, which is not the case for Infomap and Louvain. Furthermore, Demon outputs different number of communities found for the same network. For example, we applied the Demon algorithm on the stock network two times, the first time the algorithm outputs 1532 nodes contained in 4 communities, the second time it gives 1531 nodes contained in 5 different communities. As there are overlapping communities, there are many ways to plot the results. An example result when there are 5 communities is shown in Figure 12. We can observe that there is a big community in yellow, the green community is second in size is also noticeable, and other communities are very small and insignificant.

[7]https://github.com/GiulioRossetti/demon

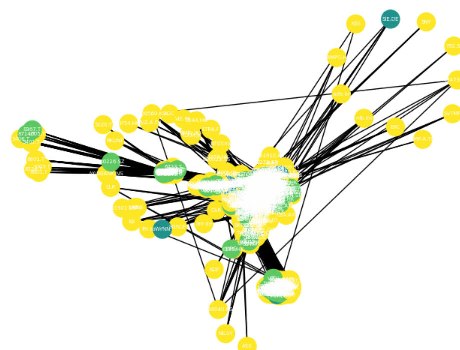

Fig. 12. Communities discovery using Demon for the stock network.

Comparing the results of three algorithms, we see that Louvain can discover the underlying community structure of the network similarly to the "country" attribute. The outputs of Infomap and Demon are hard to interpret, and hence we need further investigations into the nodes of the discovered communities to understand better their results.

C. Tie Strength

Another question we have considered as a part of our investigation was the impact of strong and weak ties on the connectedness and resilience of the network. As a measure of tie strength, we have employed Jaccard coefficient J, which is defined between any two nodes u and v of the network as:

$$J(u,v) = \frac{|\Gamma(u) \cap \Gamma(v)|}{|\Gamma(u) \cup \Gamma(v)|}, \quad (4)$$

where $\Gamma(u)$ is the set of neighbors of node u. Jaccard coefficient takes values between 0 and 1, with larger values corresponding to stronger connections between the two nodes. We define *strong ties* as pairs of nodes with a large Jaccard coefficient that also have an edge between each other. On the other hand *weak ties* are pairs of nodes with large Jaccard coefficient, which are not directly connected by an edge.

To analyze impact of strong ties, we simulate a targeted attack on the network, where edges are removed in order of decreasing Jaccard coefficient. As for analysis of weak ties, we instead simulate a targeted attack on the network, where pairs of unconnected nodes are removed in order of decreasing Jaccard coefficient. In this investigation, we adopt the probability of finding a random node of the network in the giant component P_∞ as the measure of network resilience, and as a measure of network connectedness we use the average clustering coefficient $\langle C \rangle$. As the network is destroyed, we record these quantities as a function of the fraction of the nodes removed from the network $f = 1 - \frac{L}{L_{\text{init}}}$. Results of these procedures are shown in the following figures.

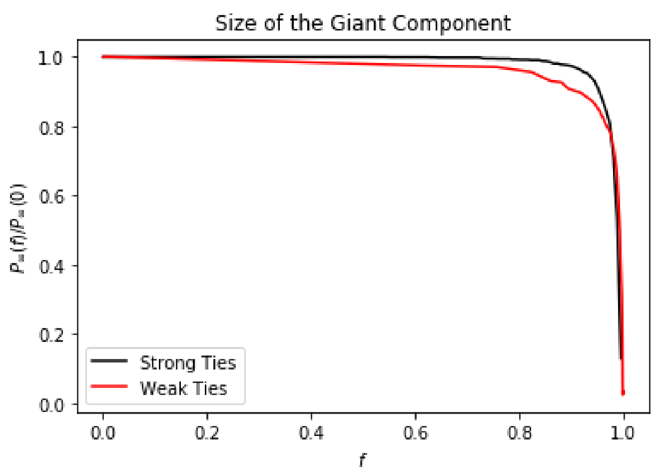

Fig. 13. Impact of strong and weak ties on network resilience.

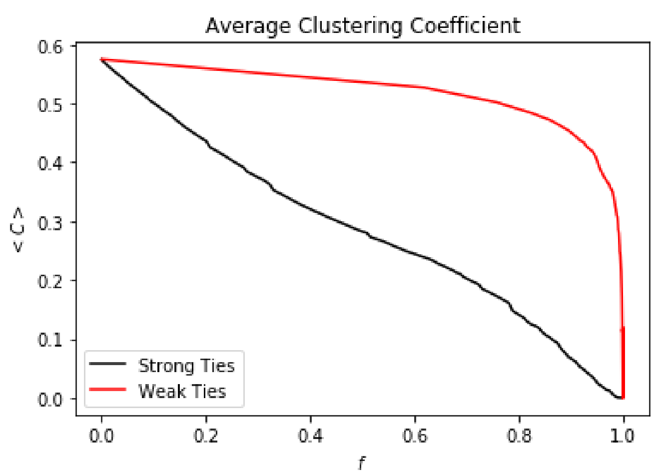

Fig. 14. Impact of strong and weak ties on network connectedness.

From Figure 13 we can note quite surprisingly that sequential removal of weak ties seem to have a greater effect on the breakdown of the giant component, than removal of strong ties, most prominently observed as f surpasses 0.8. On he other hand, when it comes to network connectedness as seen in Figure 14, removal of strong ties leads to approximately linear decrease of $\langle C \rangle$ to zero, while removal of weak ties has small effects up until around $f = 0.9$, when $\langle C \rangle$ starts to drop exponentially.

D. Threshold Model

In this task, we accessed the ability of the three networks to survive random failures or targeted attacks, using the inverse percolation theory. Let f be the ratio of nodes to be removed from a network, the threshold f_c be the value of f where the giant component of the network dissolves and the remaining nodes consists of many tiny components. According to the Molloy-Reed criterion [5], we have

$$f_c = 1 - \frac{1}{\frac{\langle k^2 \rangle}{\langle k \rangle} - 1}, \tag{5}$$

where $\langle k \rangle = \sum_{k_i} k_i p(k_i)$ is the average degree of the network, and $\langle k^2 \rangle = \sum_{k_i} k_i^2 p(k_i)$. Figure 15 plots the values of the removed nodes ratio f versus P_∞, the probability that a randomly chosen node belongs to the largest cluster, for the stock values correlation network.

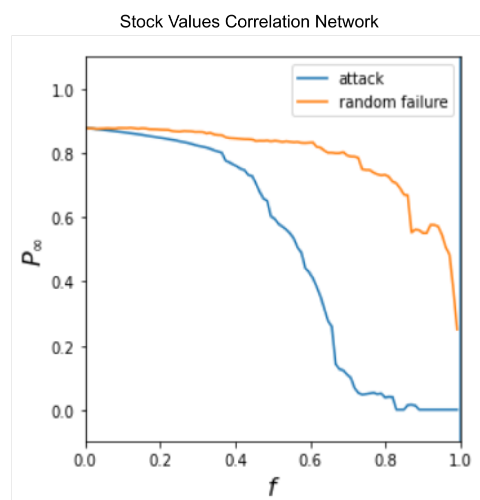

Fig. 15. Threshold model for the stock correlation network. "attack": nodes with the highest degrees are removed first. "random failure": nodes are removed randomly.

Similarly, the threshold models for the ER network and BA network are shown in Figure 16 and 17, respectively. We can see that the thresholds f_c are almost 1 for all three models, showing that they are very robust to targeted attacks and even more so to random failures. For the stock correlation network, the probability that a node belongs to a big cluster never goes down close to 0% until more than 80% of the nodes are removed. This may be explained by the fact that the network is highly interconnected. For BA and ER networks, the largest cluster only dissolves after almost all nodes are removed.

IV. Conclusion

In conclusion, we have performed analysis of a stock correlation network composed of 1955 node entries taken from Forbes Global 2000 list. We have compared our network with synthetic Erdős–Rényi and Barabási–Albert networks that had comparable number of nodes and edges. We have determined that our network overall had a higher degree of clustering, a scale-free degree distribution, and the largest betweenness

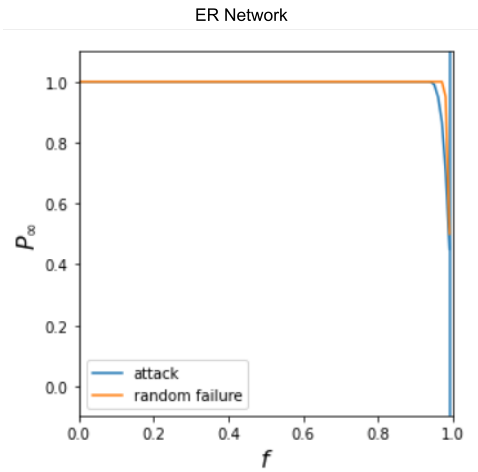

Fig. 16. Threshold model for the ER network.

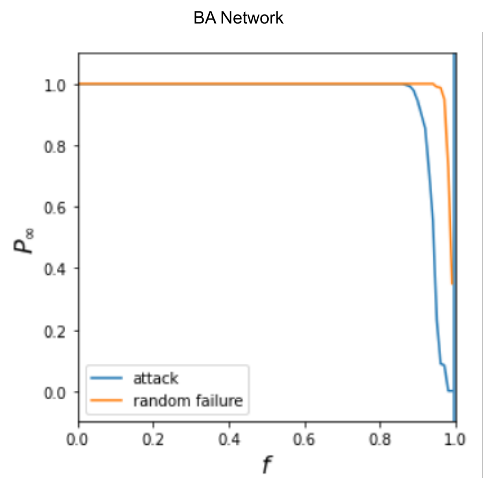

Fig. 17. Threshold model for the BA network.

centrality in comparison to other models. We then performed community detection routines on our network, mainly Louvain, Infomap and DEMON algorithms, and compared the communities detected with these methods to country-based network partitioning. We have assessed an impact of strong and weak ties on network resilience and connectedness using Jaccard coefficient and found that removal of weak ties has a slightly greater effect on the size of the network's giant component, while removal of strong ties has a bigger impact on reduction of the network's average clustering coefficient. Finally, we have evaluated a threshold model on all three networks and found all of them to be highly robust.

There are still quite a few ventures one could explore by analysing stock correlation networks. For instance, adding the type of industry attribute to each node, one could assess cross-industrial collaboration trends in the economy. Similarly, one could assess collaboration between different countries by finding the average path length between node clusters grouped by country. Furthermore, creating a temporal network by considering time-dependent correlation (i.e. computing correlation dynamically as the new opening market values arrive every day), and then analyse trends in edge formation between nodes can potentially produce a predictive model as well. In our work we have barely scratched the surface of econometrics. Nonetheless, in the course of this exploration we have witnessed first-hand the importance of the study of complexity in the context of global economics.

References

[1] C. E. Shannon. "A mathematical theory of communication". In: *The Bell System Technical Journal* 27.3 (1948), pp. 379–423. DOI: 10.1002/j.1538-7305.1948.tb01338.x.

[2] Vincent D Blondel et al. "Fast unfolding of communities in large networks". In: *Journal of Statistical Mechanics: Theory and Experiment* 2008.10 (Oct. 2008), P10008. ISSN: 1742-5468. DOI: 10.1088/1742-5468/2008/10/p10008. URL: http://dx.doi.org/10.1088/1742-5468/2008/10/P10008.

[3] M. Rosvall and C. T. Bergstrom. "Maps of random walks on complex networks reveal community structure". In: *Proceedings of the National Academy of Sciences* 105.4 (Jan. 2008), pp. 1118–1123. ISSN: 1091-6490. DOI: 10.1073/pnas.0706851105. URL: http://dx.doi.org/10.1073/pnas.0706851105.

[4] Michele Coscia et al. *DEMON: a Local-First Discovery Method for Overlapping Communities*. 2012. arXiv: 1206.0629 [cs.DS].

[5] Albert-László Barabási and Márton Pósfai. *Network science*. Cambridge: Cambridge University Press, 2016. ISBN: 9781107076266 1107076269. URL: http://barabasi.com/networksciencebook/.