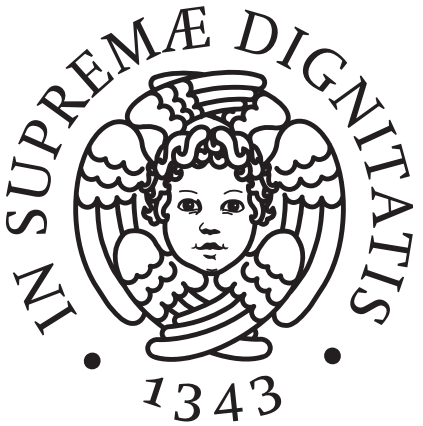


Università di Pisa

Social Network Analysis
A.A. 2017/2018

Cambridge Analytica and Facebook: The Scandal and the Fallout on Twitter

Gianmarco Ricciarelli 555396
Stefano Carpita 304902

Data drives all we do.

Cambridge Analytica main slogan.

Rules don't matter for them.
For them, this is a war, and it's all fair.

Christopher Wylie,
former datascientist at Cambridge Analytica, about its leaders.

Contents

1 | The case story

On Saturday 17 of March 2018, the newspapers The Observer and The New York Times broke reports on how the consulting firm Cambridge Analytica harvested private information from the Facebook profiles of more than 50 million users without their permission, making it one of the largest data leaks in the social network's history. [1]. REF OBSERVER

The whistleblower Christopher Wylie, datascientist and former director of research at Cambridge Analytica revealed... Cambridge Analytica described itself as a company providing consumer research, targeted advertising and other data-related services to both political and corporate clients.

What, Where, Who, Why, Where ?

Timeline da sistemare: [2]

- March 17, 2018: The Observer and The New York Times publish joint reports on data harvesting by Cambridge Analytica. UK Information Commissioner Elizabeth Denham issues statement that they are "investigating circumstances in which Facebook data may have been illegally acquired and used." Politicians in US and UK call for investigation.
- March 19, 2018: Channel 4 News publishes part 1 of their undercover investigation into Cambridge Analytica. Facebook sends investigators to Cambridge Analytica's offices. UK Information Commissioner orders them to stand down.
- March 20, 2018: Channel 4 News publishes part 2 of their undercover investigation into Cambridge Analytica, where they boast about getting Donald Trump elected. British MP Damian Collins calls on Facebook to present oral evidence on Cambridge Analytica. Facebook agrees to send former operations manager Sandy Parakilas. Facebook holds internal Q&A with attorney Paul Grewal to discuss the crisis, but CEO Mark Zuckerberg and COO Sheryl Sandberg do not attend. Cambridge Analytica suspends CEO Alexander Nix. Facebook demands to inspect Christopher Wylie's phone. FTC opens investigation into Facebook.
- to be continued...

2 | Building the network

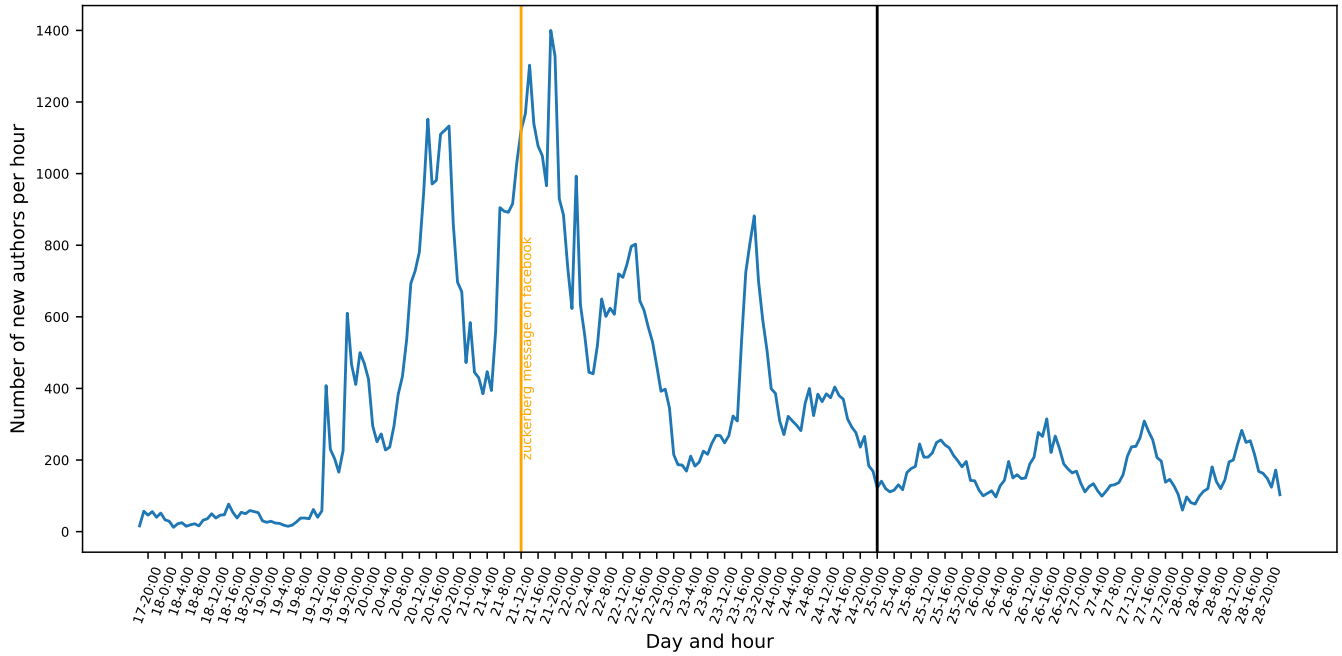


Figure 2.1: New authors time history

3 Network properties

3.1 Degree distribution

	g	g_und	g_er	g_ba
L	2501757	1895878	4318406	4989628
N	65729	65729	65729	65729
density	0.00058	0.00088	0.001	0.00231
gamma	2.6	None	None	2.9
gamma_in	2.4	None	None	None
gamma_out	2.9	None	None	None
gamma_tot	2.6	None	None	None
k_avg	38	57	65	151
k_in_max	19064	None	109	None
k_in_min	0	None	36	None
k_max	19073	19065	183	3640
k_min	1	1	84	76
k_out_max	4130	None	103	None
k_out_min	0	None	35	None

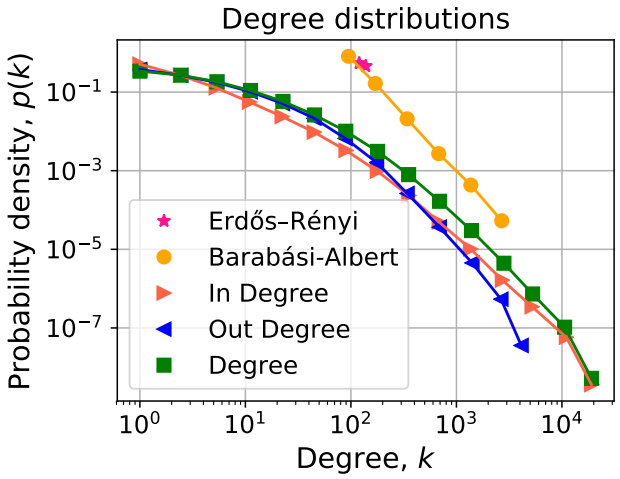


Figure 3.1

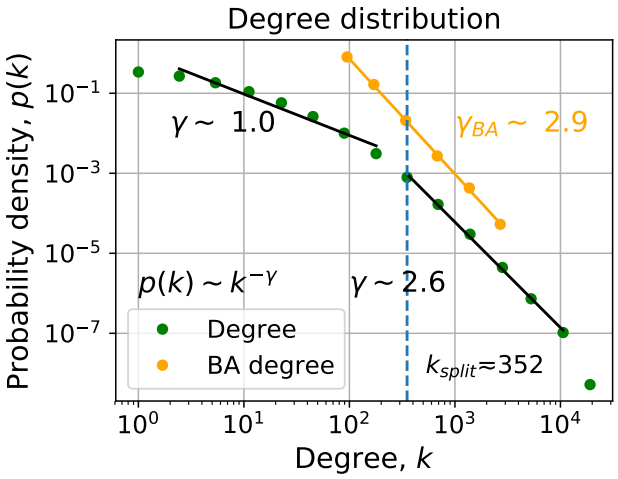


Figure 3.2

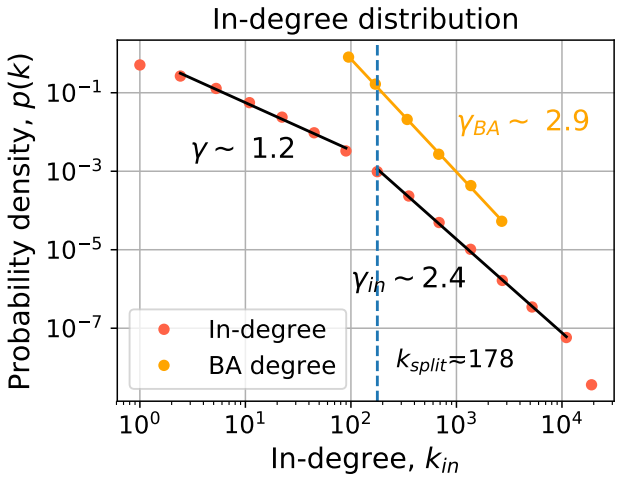


Figure 3.3

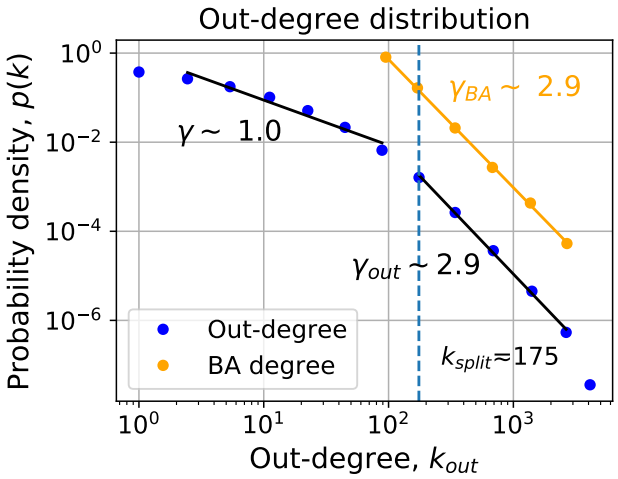


Figure 3.4

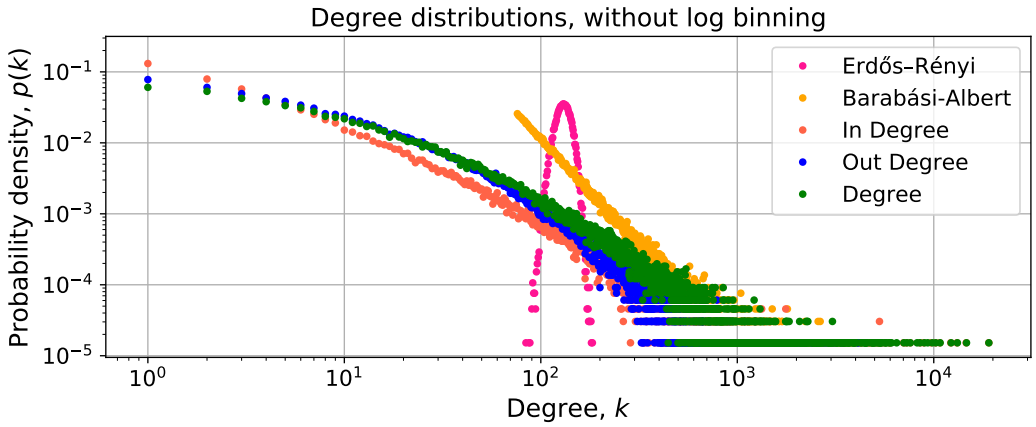


Figure 3.5: New authors time history

3.1.1 Random graphs

In order to generate an Erdos-Renyi random network we have choosen a "linking probability" p using the average degree of the original undirected network, by using eq. 3.1.

$$p_{ER} \approx \frac{\langle k \rangle}{N} = \frac{57}{65729} \approx 0.001 \tag{3.1}$$

Each new node of the random network generated with the Barabasi-Albert model has been attached to the other nodes with a number of links m equal to the average degree of the original network, considered undirected:

$$m = 2 \langle k \rangle = 76 \tag{3.2}$$

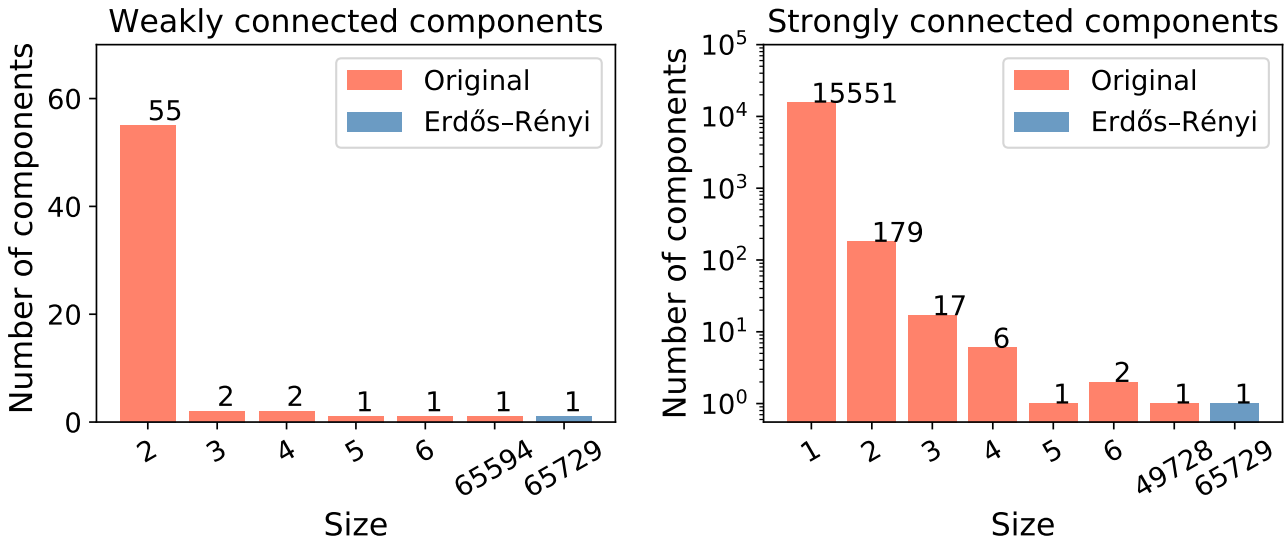


Figure 3.6: Connect components

3.2 Path analysis

In order to exactly estimate the average path length $\langle d \rangle$ it would be necessary to compute all the node-node distances of the network. These procedure results infeasible with the computation resources available, as shown in Fig. 3.14. In real networks the path length distribution is quite close to a normal distribution, as shown in [3]. The average path length has then been estimated statistically, random sampling a number n of node pairs, sufficient to achieve a narrow confidence interval for the mean. The assumption of normality of the distribution it is strong, but not necessary. The convergence of the computed mean to the expected value is guaranteed by the central limit theorem with the assumptions that the distances are independent, identically distributed, and with finite variance. The average path length has been estimated by the average of the distances D_i for each sampled node pair, and computing its standard deviation:

$$\langle d \rangle = \frac{\sum D_i}{n} \ , \ \sigma(\langle d \rangle) = \frac{s}{\sqrt{n}} \tag{3.3}$$

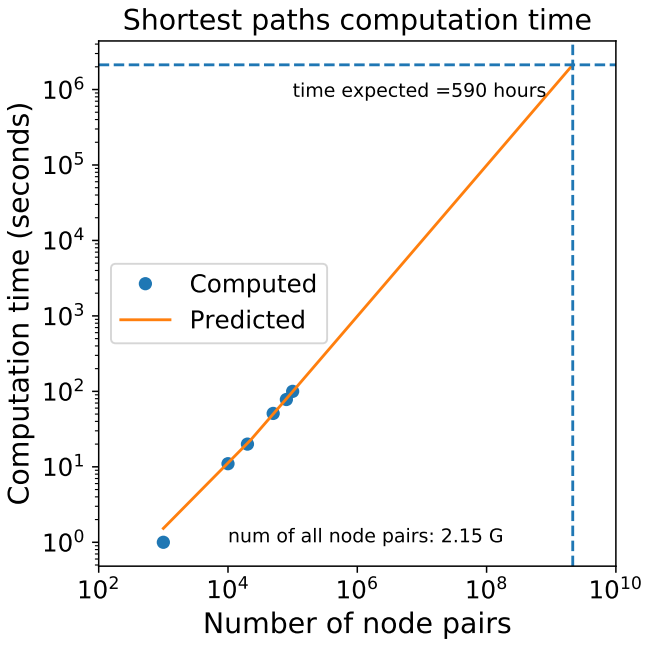


Figure 3.7: Shortest paths computation time by number of pairs

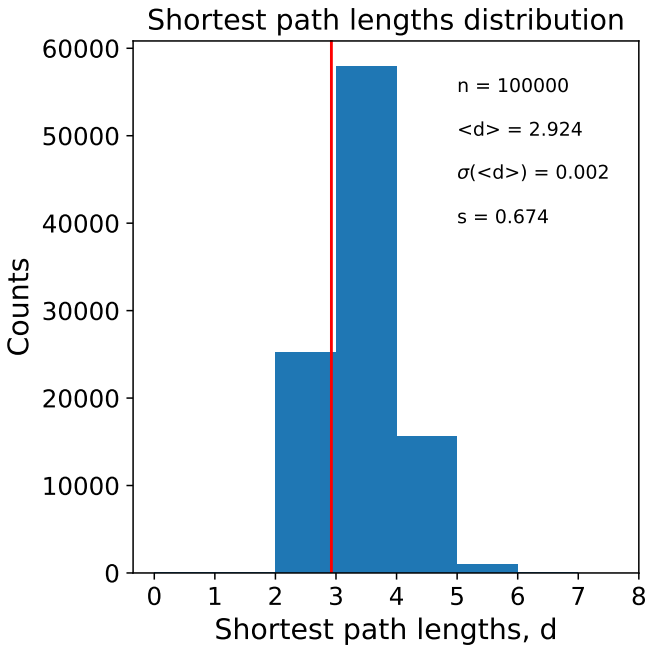


Figure 3.8: Shortest paths distribution

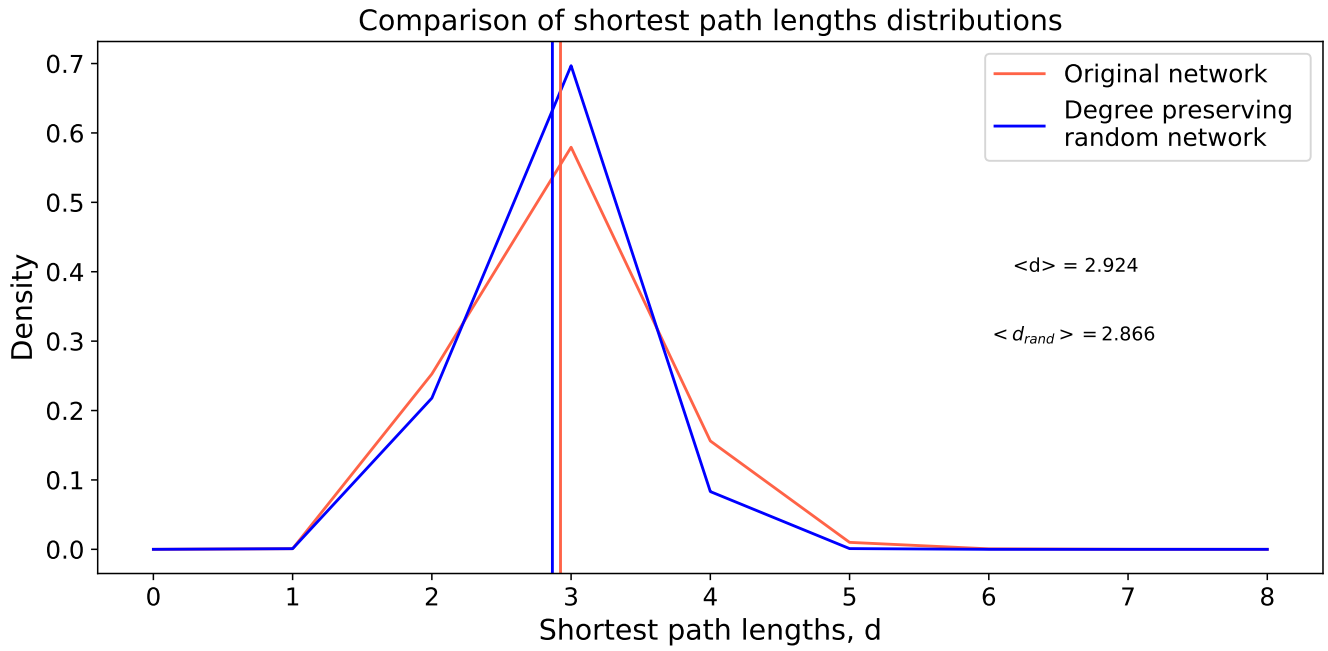


Figure 3.9: Shortest paths distributions comparison between the original largest connected component and a random network with degree preservation.

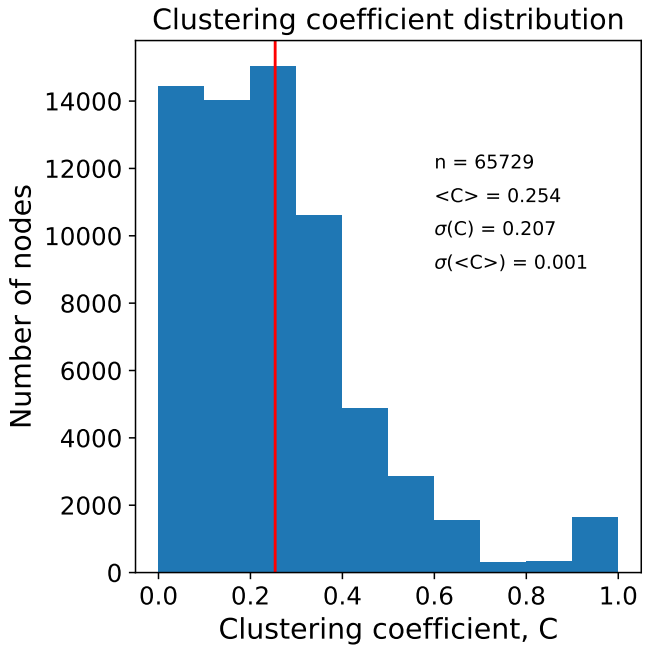


Figure 3.10: Clustering coefficient distribution

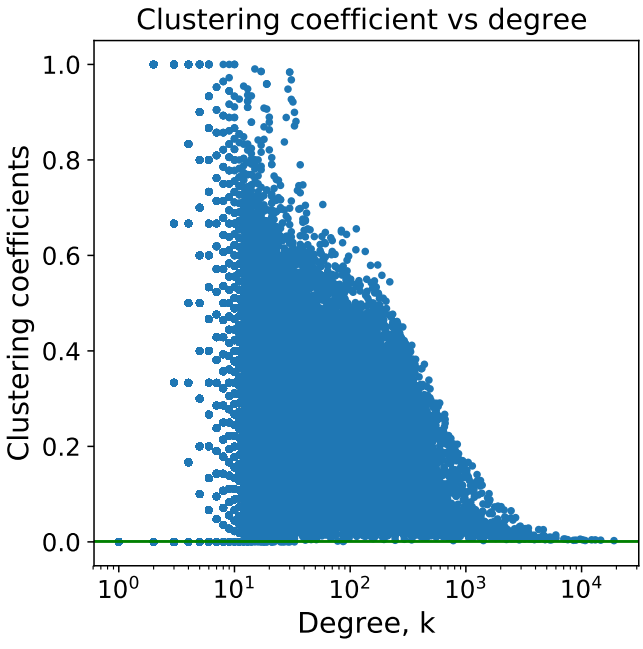


Figure 3.11: Clustering coefficients as function of the degree

3.3 Hubs analysis

The hubs of the crawled social network of tweets authors about the Cambridge Analytica-Facebook scandal are mainly news mass media, as expected. In Fig. 3.12 the 30 biggest hubs are represented by indicating the in-degree of the crawled network, corresponding to the number of authors following the hub, versus the actual total number of followers on Twitter. The "The New York Times" is the biggest hub, with the maximum number of both in-degree and number of followers. We observe that there is an obvious positive correlation between in-degree and followers, with some variations. In particular, let's take a pair of hubs having similar followers count, such as the "Washington Post" and the "Huffington Post". The "Washington Post" has a larger in-degree than the second. This difference can be interpreted as a larger interest in the scandal from the people following the "Washington Post" respect to the ones following the "Huffington Post". We can define a quantity to measure this interest:

$$\text{Interest} \equiv \frac{\text{in-degree}}{\#\text{followers}} \tag{3.4}$$

This measure represents the percentage of followers that being interested in the scandal had published a tweet about the subject.

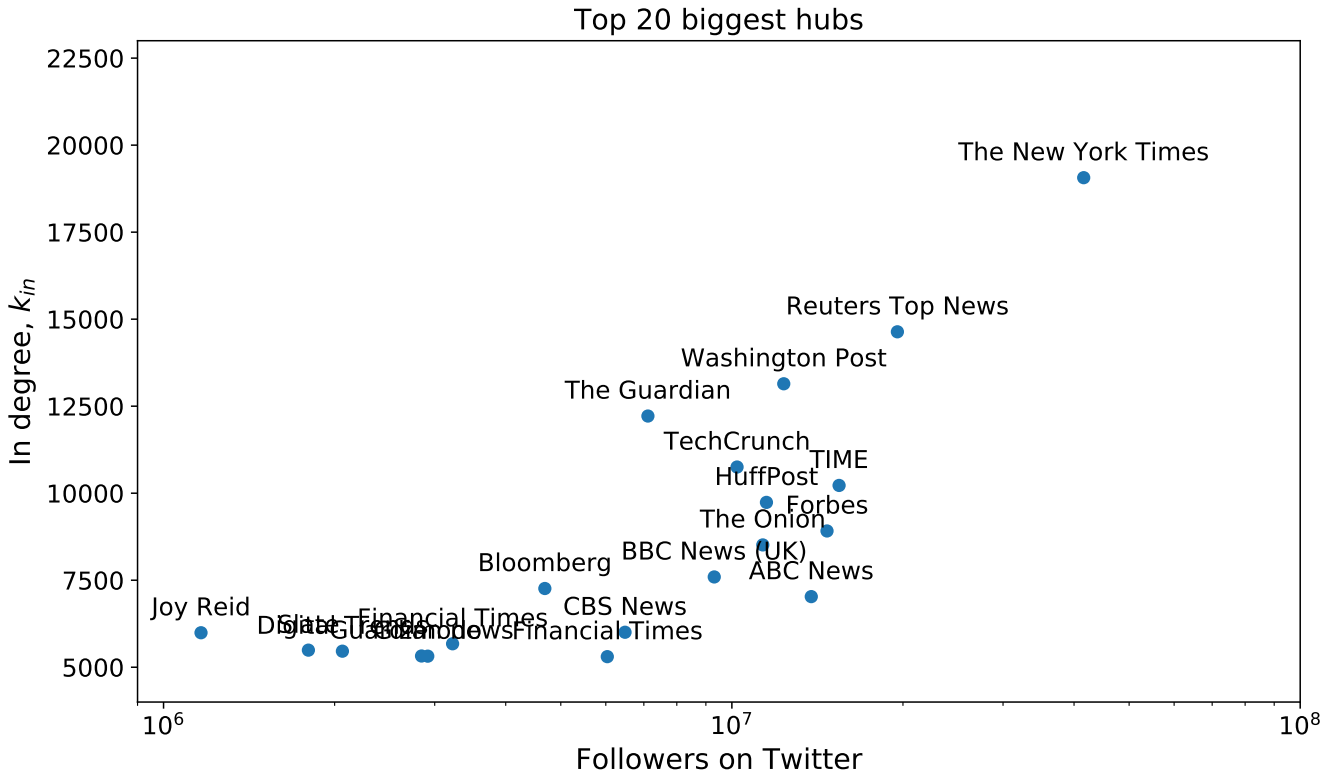


Figure 3.12

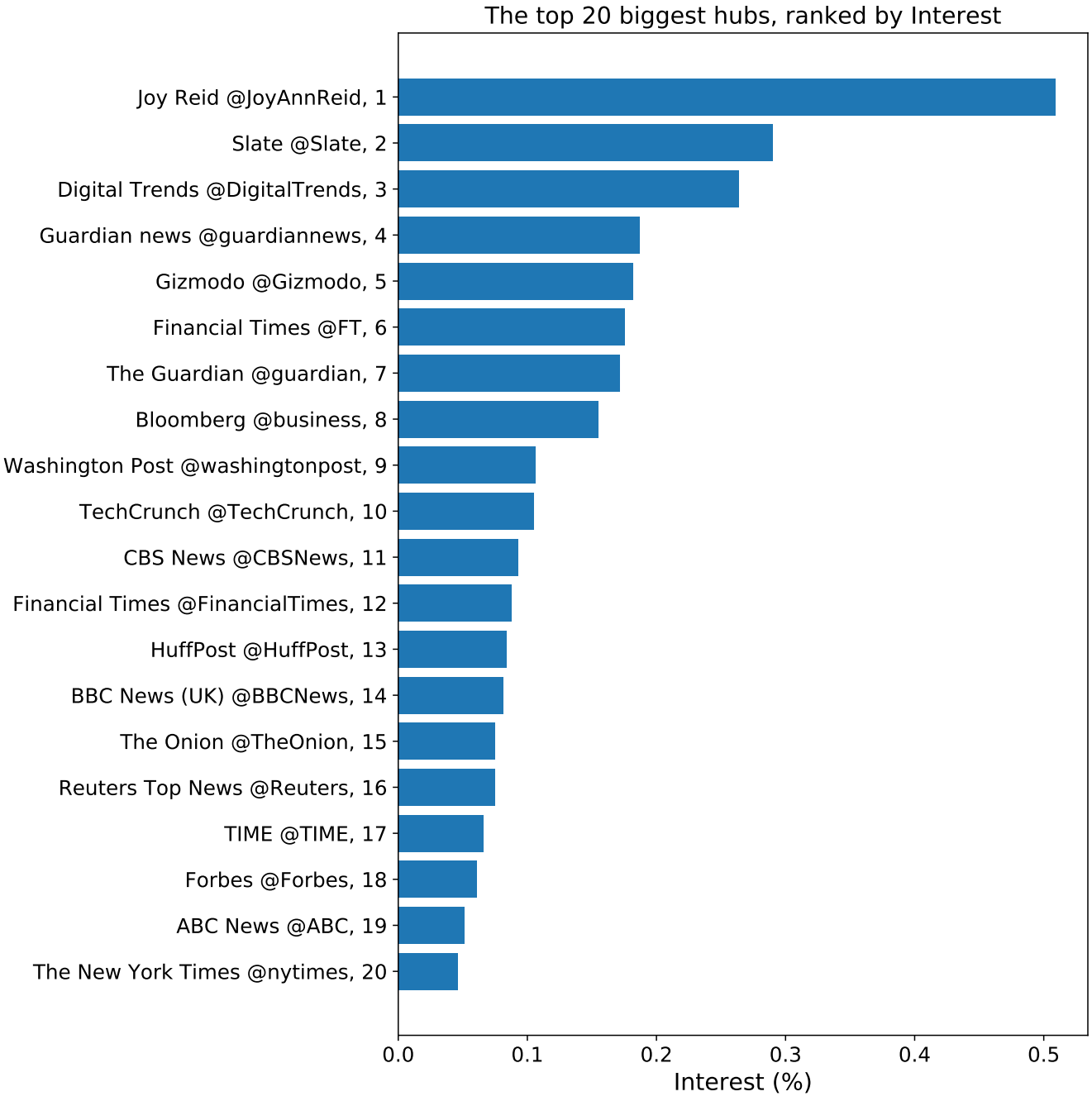

The top 20 biggest hubs, ranked by Interest
Joy Reid @JoyAnnReid, 1
Slate @Slate, 2
Digital Trends @DigitalTrends, 3
Guardian news @guardiannews, 4
Gizmodo @Gizmodo, 5
Financial Times @FT, 6
The Guardian @guardian, 7
Bloomberg @business, 8
Washington Post @washingtonpost, 9
TechCrunch @TechCrunch, 10
CBS News @CBSNews, 11
Financial Times @FinancialTimes, 12
HuffPost @HuffPost, 13
BBC News (UK) @BBCNews, 14
The Onion @TheOnion, 15
Reuters Top News @Reuters, 16
TIME @TIME, 17
Forbes @Forbes, 18
ABC News @ABC, 19
The New York Times @nytimes, 20
0.0
0.1
0.2
0.3
0.4
0.5
Interest (%)

Figure 3.13

3.4 Italian sub network

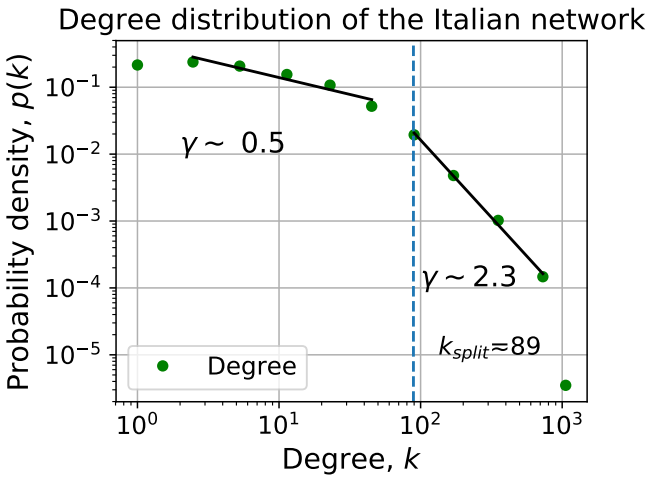


Figure 3.14: Shortest paths computation time by number of pairs

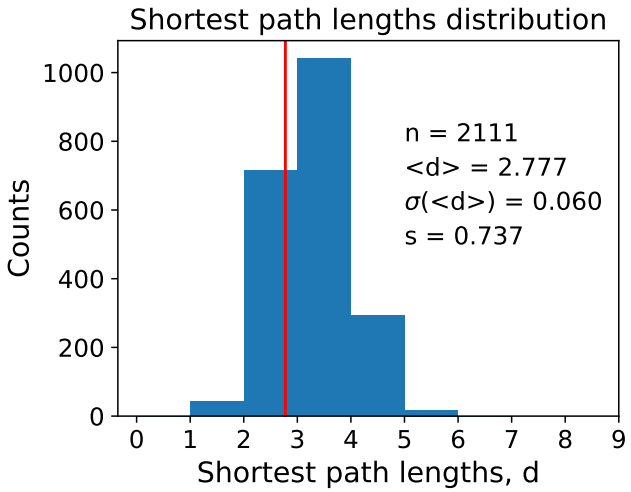


Figure 3.15: Shortest paths distribution

4 | Network dynamics

5 | Network robustness

In this chapter we'll provide some results about the **robustness** and **attack tolerance** of our network. Thaking as reference the concepts described in [4], we'll define the **critical threshold** of our network, and we'll test its robustness against attacks conducted following a random nodes' selection or one based on decreasing degree centrality. Finally we'll test the **Failure Propagation Model**, following our implementation of the model, on our network.

5.1 Critical threshold

As described in [4], we have obtained the **critical threshold** representing the fraction of the nodes that must be removed to break apart our network. This fraction, represented by f_c, is obtained by the following formula:

$$f_c = 1 - \frac{1}{\frac{\gamma-2}{3-\gamma} k_{min}^{\gamma-2} k_{max}^{3-\gamma} \ - \ 1} = 1 - \frac{1}{1.50 * 1^{0.6} * 19073^{0.4} \ - \ 1} = 0.99$$

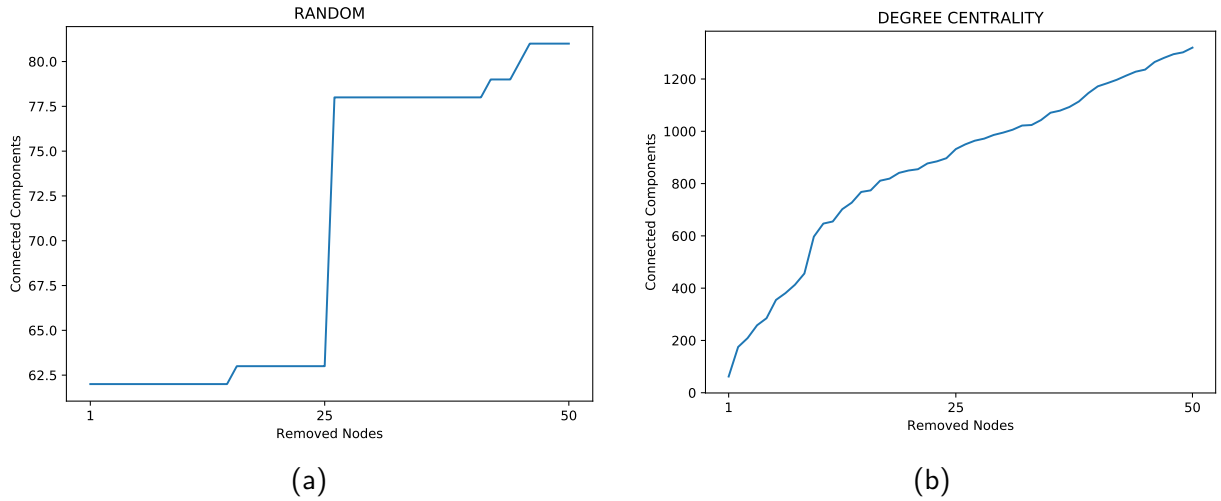
which, remembering that the γ for our scale-free network corresponds to 2.6 and that k_{min} and k_{max} are eguals to 1 and 19073 respectively, tells us that, in order to break apart our network it is mandatory to remove the 99% of the nodes. Keeping in mind that our network is, in fact, a finite network, we can adjust the obtained result by utilizing the following formula, still in [4]:

$$f_c \approx 1 - \frac{C}{N^{\frac{3-\gamma}{\gamma-1}}} \approx 1.00$$

where $C = \frac{1}{\sum_{k=1}^{\infty} k^{-\gamma}}$ is a constant and N represents the number of nodes of the network. As we can see, this new approximation tells us that in order to break apart our network the totality of its nodes must be removed.

5.2 Simulation of an attack

In order to validate the results obtained in Section 5.1, here we simulate an attack to our network. We've chosen to simulate the remotion of 50 nodes from the network following two distinct criterions: **random selection** and **degree centraility** (decreasing order). For every criterion we've monitored the fragmentation of the connected components.



(a) (b)

Figure 5.1: In Figure 5.1a we can see the fragmentation of the connected components during the remotion based on a randomic choice, while in Figure 5.1b we can see the same fragmentation, but this time based on decreasing degree centraility.

As we can see, for the randomic choice of the nodes to be removed, the structure of the network is barely altered. After the random remotion of 50 nodes, the original 62 connected components became slightly more than 80. For the remotion of the nodes based on decreasing degree centrality there is, as expected, a different situation. This kind of criterion guarantees that the original structure of the network is broken apart more easily, because the original network's hubs are removed one by one in decreasing order.

5.3 Simulation of a Failure Propagation Model

To test more the robustness of our network, we've written the code in order to implement (and test) the **Failure Propagation Model**, as described in [4]. In Figure 5.2 you can see some iterations of the model in which we used different values for the φ parameter.

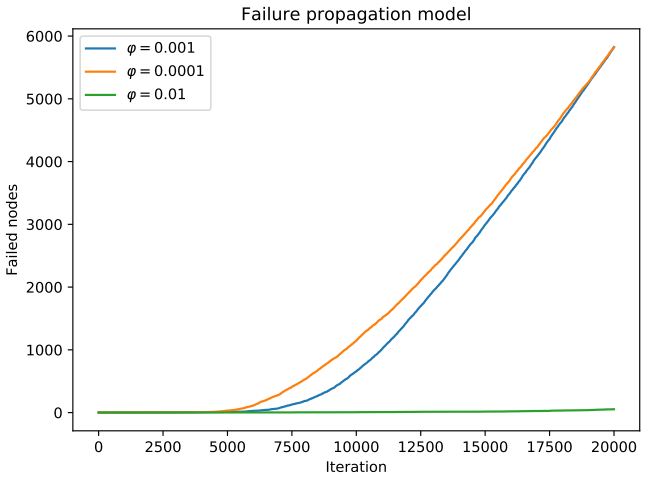


Figure 5.2: The test for the Failure Propagation Model was conducted over 20000 iterations.

We can see that, as expected, the network doesn't accuse the failure propagation for φ eguals to 0.01, in which less than 30 nodes failed. For the smaller values of φ we can see that the situation is different, with a greater amount of nodes which fail over the iterations.

6 | Communities discovery

In this chapter we'll provide the results obtained by applying **K-Clique**, **Label Propagation**, **Louvain**, **Girvan-Newman** and **Demon** to a sample of 2000 nodes taken from the original network. We've chosen to sample the crawled data in order to ease the application of the various algorithms. Each partition is evaluated by applying an implementation of the scoring functions listed in [5], and, for each algorithm, the results are represented in a table. Together with the results of the scoring functions are also provided the total number of communities discoverd (Communities), the number of nodes in the smallest/biggest community (Smallest/Biggest), the hashtags utilized in the biggest community (Tags) and finally the languages of the users in the biggest community (Langs). Finally it is also provided an evaluation for the subset of the original network composed only by italian users.

6.1 K-Clique

We have chosen to apply the **K-Clique** algorithm, described in [6], to the sample using three different values for k: 3, 4 and 5, respectively. The results are represented in Table 6.1. As we can see, even if the result is not so good, the best partition is obtained by using k eguals to 3, which returns a low modularity partition composed by low degree nodes. It is interesting to note that the biggest partitions returned by the first application of the algorithm is composed by english and deutsch speaking users, while the other iterations returns community composed only by english speaking users.

K	Communities	Biggest	Smallest	Modularity	Conductance	IED	AND	Tags	Langs
3	29	51	3	0.14	0.62	0.20	2.72	zuckerberg, cambridgeanalytica, deletefacebook, facebook, facebookgate	English, Deutsch
4	13	14	4	0.020	0.68	0.21	4.23	cambridgeanalytica, deletefacebook, facebook	English
5	5	12	5	0.011	0.63	0.23	4.83	cambridgeanalytica, deletefacebook, facebook	English

Table 6.1: Evaluation of the partitions obtained by the application of the K-Clique algorithm.

6.2 Label Propagation

In Table 6.2 are represented the results of the application of the **Label Propagation** algorithm, described in [7]. According to the modularity score the partition provided by this algorithm represent a good subdivision of the original network, even if it is composed for the vast majority by small communities, as suggested by the high number of communities and the low value for the Average Node Degree score. Since this partition is composed by an high number of communities, the biggest community aggregates a vast variety of languages.

Communities	Biggest	Smallest	Modularity	Conductance	IED	AND	Tags	Langs
1278	136	1	0.68	0.28	0.19	1.38	cambridgeanalytica, deletefacebook, privacy, zuckerberg, facebook, facebookgate	Various

Table 6.2: Evaluation of the partition obtained by the application of the Label Propagaion algorithm.

6.3 Louvain

The application of the **Louvain** algorithm, described in [8], along with the iteration of the Label Propagation algorithm, returns the best partition among all the partitions returned by the other algorithms. The results of its application are represented in Table 6.3. As for the Label Propagation algorithm, this partition also is composed by an high number of small communities, and the biggest community is composed by users who speaks various languages.

Communities	Biggest	Smallest	Modularity	Conductance	IED	AND	Tags	Langs
1193	133	1	0.76	0.042	0.17	1.60	cambridgeanalytica, deletefacebook, privacy, zuckerberg, facebook, facebookgate	Various

Table 6.3: Evaluation of the partition obtained by the application of the Louvain algorithm.

6.4 Girvan-Newman

For the **Girvan-Newman** algorithm, described in [9], we've decided to record the results of 5 iterations over the sample network. The results are represented in Table 6.4. As you can see, the first three iterations returns very poor partitions, with low modularity scores, due to the fact that the edges with the highest betweenness centrality selected in the various iterations doesn't provide a good grade of separation among the nodes of the network. With the fourth and fifth iterations, there is a consistent improvement either in the modularity score and in the other measures.

Iteration	Communities	Biggest	Smallest	Modularity	Conductance	IED	AND	Tags	Langs
1	1181	703	1	0.17	0.00098	0.22	1.23	cambridgeanalytica, deletefacebook, privacy, zuckerberg, facebook, facebookgate	Various
2	1182	659	1	0.24	0.0019	0.21	1.27	cambridgeanalytica, deletefacebook, privacy, zuckerberg, facebook, facebookgate	Various
3	1183	588	1	0.38	0.0048	0.21	1.35	cambridgeanalytica, deletefacebook, privacy, zuckerberg, facebook, facebookgate	Various
4	1184	575	1	0.40	0.0081	0.20	1.34	cambridgeanalytica, deletefacebook, privacy, zuckerberg, facebook, facebookgate	Various
5	1185	458	1	0.57	0.011	0.20	1.40	zuckerberg, cambridgeanalytica, deletefacebook, facebook, facebookgate	Various

Table 6.4: Evaluation of the partition obtained by the application of the Girvan-Newman algorithm.

In general, the partitions returned by the five iterations of the algorithms are all composed by an high number of small communities, with the biggest community growing smaller iteration after iteration. This kind of fragmentation, as seen before, for every iteration produces a biggest community with diffent types of users.

6.5 Demon

Finally in Table 6.5 we provide the results of the application of the **Demon** algorithm, described in [10], that we tested for five different values of ϵ, 0.10, 0.25, 0.50, 0.75 and 0.90, respectively. In general, the five partitions are not so good from the point of view of the modularity score, with the third application of the algorithm beign the best. Contrary to the results obtained by the application of the other algorithms, the partitions for the Demon algorithm are all composed by a small number of communities.

Epsilon	Communities	Biggest	Smallest	Modularity	Conductance	IED	AND	Tags	Langs
0.10	10	147	4	0.07	0.46	0.082	4.41	zuckerberg, cambridgeanalytica, deletefacebook, facebook, facebookgate	Various
0.25	11	63	4	0.095	0.44	0.094	4.31	zuckerberg, cambridgeanalytica, deletefacebook, facebook, facebookgate	English, Deutsch
0.50	21	43	4	0.11	0.57	0.10	4.15	zuckerberg, cambridgeanalytica, deletefacebook, facebook, facebookgate	English, Deutsch
0.75	39	25	4	0.068	0.62	0.12	3.96	zuckerberg, cambridgeanalytica, deletefacebook, facebook, facebookgate	English, Deutsch
0.90	89	24	4	0.071	0.68	0.15	3.65	zuckerberg, cambridgeanalytica, deletefacebook, facebook, facebookgate	English, Deutsch

Table 6.5: Evaluation of the partition obtained by the application of the Demon algorithm.

6.6 Comparisons

In this section of the chapter we compare the algorithms used so far by confronting the best instances among the iterations provided in the past sections. The comparisons are performed by using the NF1 score, as described in [11].

A1	A2	F1 mean	Ground Truth Communities	Identified Communities	Community Ratio	Ground Truth Matched	Node Coverage	NF1
K-Clique 3	Label Propagation	0.44	714.0	21.0	0.029	0.022	0.10	0.0076
K-Clique 3	Louvain	0.32	683.0	21.0	0.031	0.016	0.10	0.0027
K-Clique 3	Girvan-Newman 5	0.25	677.0	21.0	0.031	0.012	0.10	0.0011
K-Clique 3	Demon 0.50	0.94	9.0	21.0	2.33	0.78	1.44	0.24
Label Propagation	Louvain	0.94	1193.0	1278.0	1.07	1.0	1.0	0.87
Label Propagation	Girvan-Newman 5	0.93	1185.0	1278.0	1.08	1.0	1.0	0.86
Label Propagation	Demon 0.50	0.26	21.0	1278.0	60.86	0.76	10.24	0.0025
Louvain	Girvan-Newman 5	0.99	1185.0	1193.0	1.0068	1.0	1.0	0.99
Louvain	Demon 0.50	0.32	21.0	1193.0	56.81	0.62	10.24	0.0021
Girvan-Newman 5	Demon 0.50	0.36	21.0	1185.0	56.43	0.24	10.24	0.00036

Table 6.6: Comparisons among the best iterations of the algorithms utilized in this chapter.

In Table 6.6 we can see the comparisons among the best iterations of the algorithms utilized during the community discovery phase. As we can see by the results, the best comparisons are the ones between K-Clique and Demon , with, respectively, $k = 3$ and $\epsilon = 0.50$, between Label Propagation and Louvain/Girvan-Newman (fifth iteration) and finally between Louvain and Girvan-Newman (fifth iteration).

6.7 The italian subgraph

In order to better understand the composition of our network, we selected among the nodes only the ones composed by **italian users**, and on this subgraph we've applied the same algorithms described in the past sections. The obtained partitions make clear to us that the italian users inside our network aren't disposed into a community-like shape. The best obtained result is the one returned by the Louvain algorithm, which return a modularity score of 0.26 and a partition composed by 6 communities, the biggest and the smallest being composed by 817 and 2 users, respectively. Being the partition composed by only 6 communities, the average node degree is bigger (≈ 17) that the ones returned by the other evaluations.

7 | Spreading

In this chapter we'll describe the results we obtained by applying the **SI**, **SIS**, **SIR**, and **Threshold** diffusion models both on the crawled data and on the synthetic graphs (Erdős–Rényi and Barabási–Albert) generated from the original one. In each section, a comparison between the three networks will be provided along with some details on the implementation of the tests of every model.

7.1 SI model

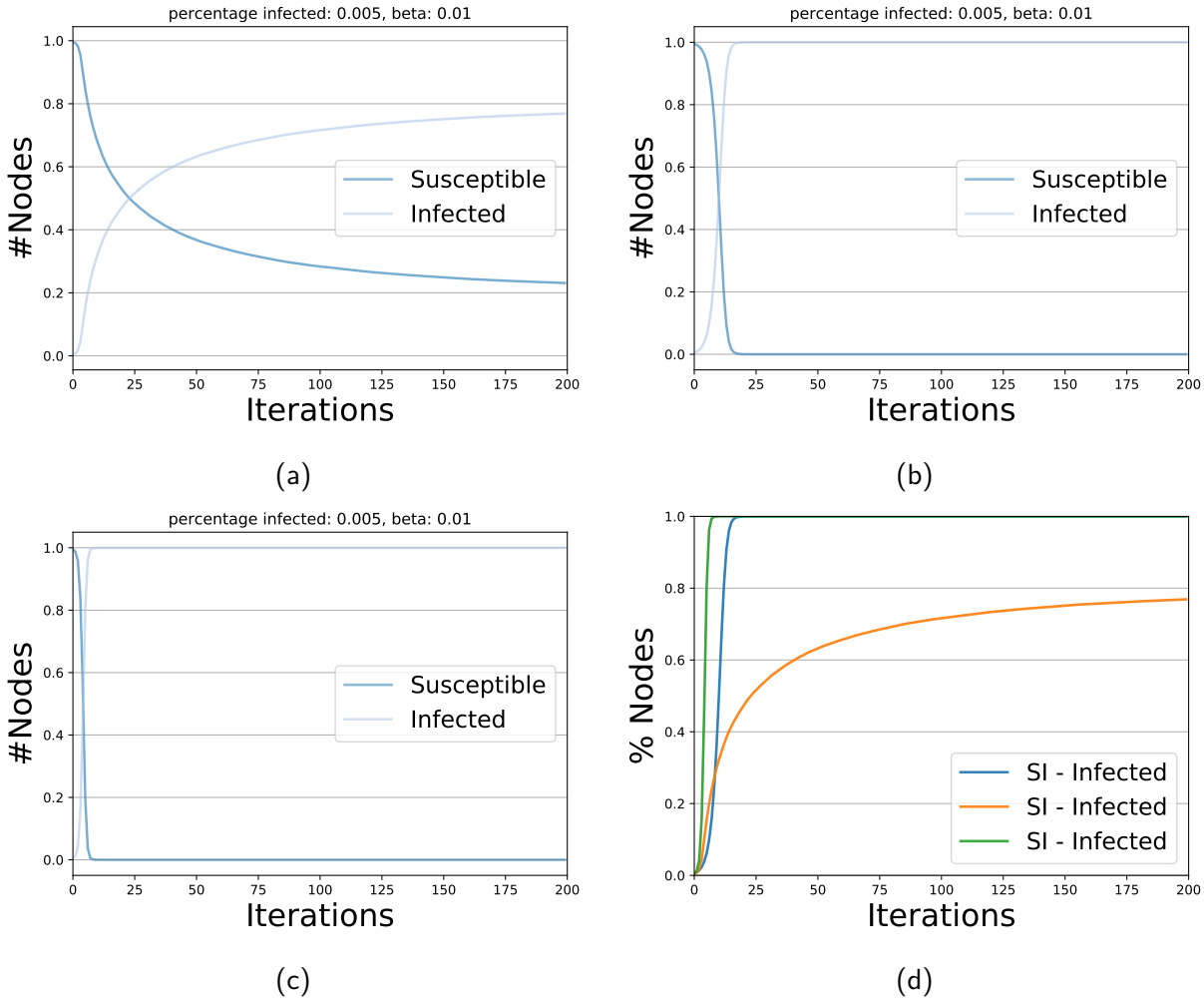


Figure 7.1: In Figure 7.1a we can see the diffusion graph for the original network, while in Figure 7.1b and in Figure 7.1c we can see the diffusion graph for the Erdős–Rényi and Barabási–Albert networks, respectively. In Figure 7.1d we can see a comparison between the infection rate of the three networks.

For the **Susceptible-Infected** model we've started with a 0.005% of the total population (3 nodes) of each network being infected, and we've choosed a value of 0.01 for the infection rate β. As you can see from Figure 7.1, the original network is the only one that doesn't reach the saturation regime, while the other networks reach it within the first 25 iterations of the model. This is due to the fact that both the Erdős–Rényi and the Barabási–Albert network are extremely connected, hence it is more easy for the infection to spread among the nodes.

7.2 SIS model

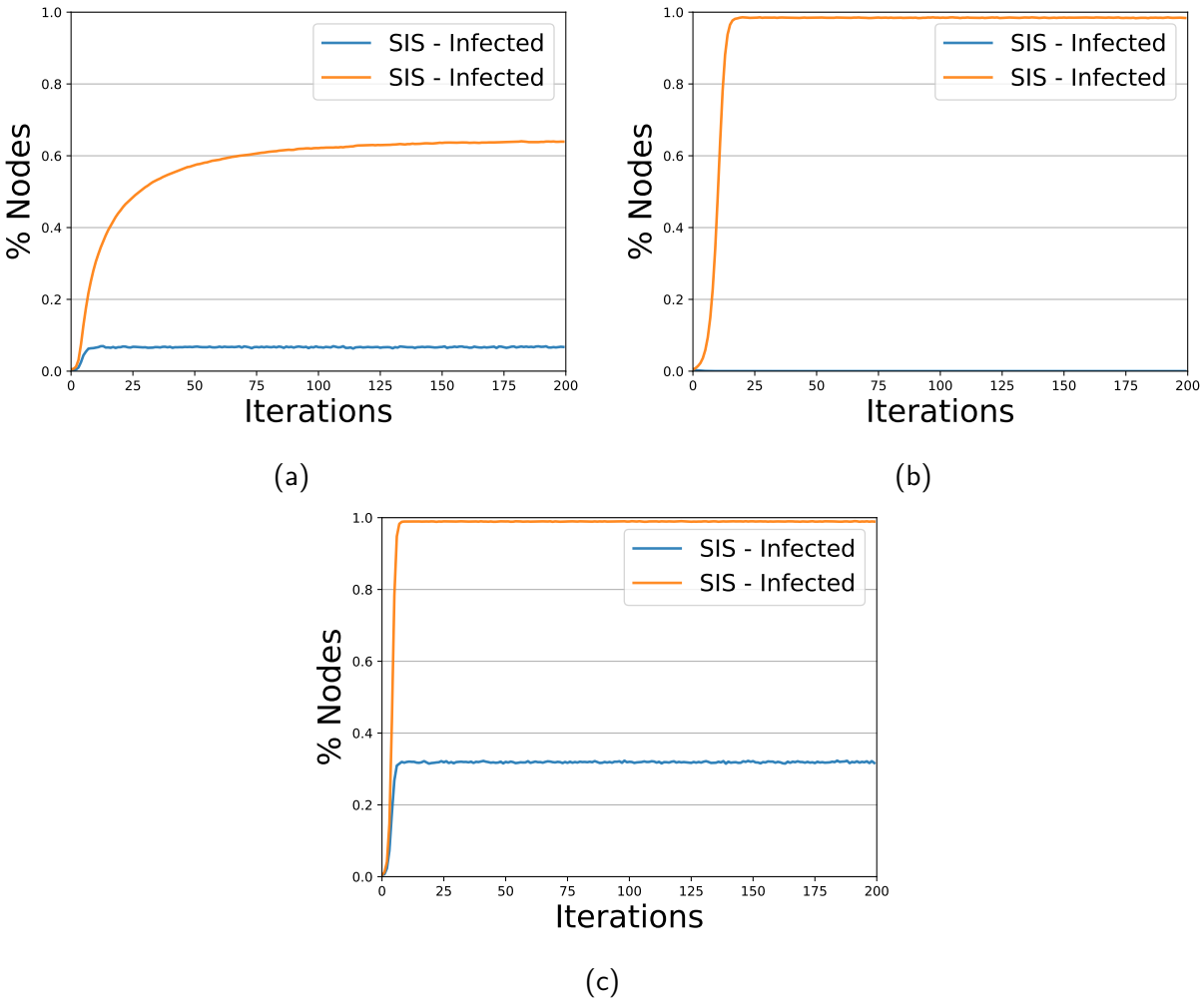


Figure 7.2: In Figure 7.2a we can see the comparison between the endemic state, in orange, and the disease free state, in blue, for the original network. The same comparison can be observed for the Erdős–Rényi and the Barabási–Albert network, respectively, in Figure 7.2b and 7.2c

For the **Susceptible-Infected-Susceptible** model, thanks to the introduction of the recovery rate μ, we can model two possible outcomes for the epidemic: the **endemic state**, characterized by a low recovery rate and by the fraction of infected individuals that follows a logistic curve similar to the one observed for the SI model, for which $\mu < \beta\langle k\rangle$, and the **disease free** state, characterized by a sufficiently high recovery rate, for which $\mu > \beta\langle k\rangle$. A comparison between this two states is represented for every network in Figure 7.2.

7.3 SIR model

The key characteristic of the **Susceptible-Infected-Recovered** model consist in introducing the probability γ for the individuals to recover from the disease and hence to be "removed" from the population instead of returning to the susceptible state. We have choosen to test this model either for the case in which γ is smaller than β and the other way around. The graphs representing this different situations for all the three networks are visible in Figure 7.3.

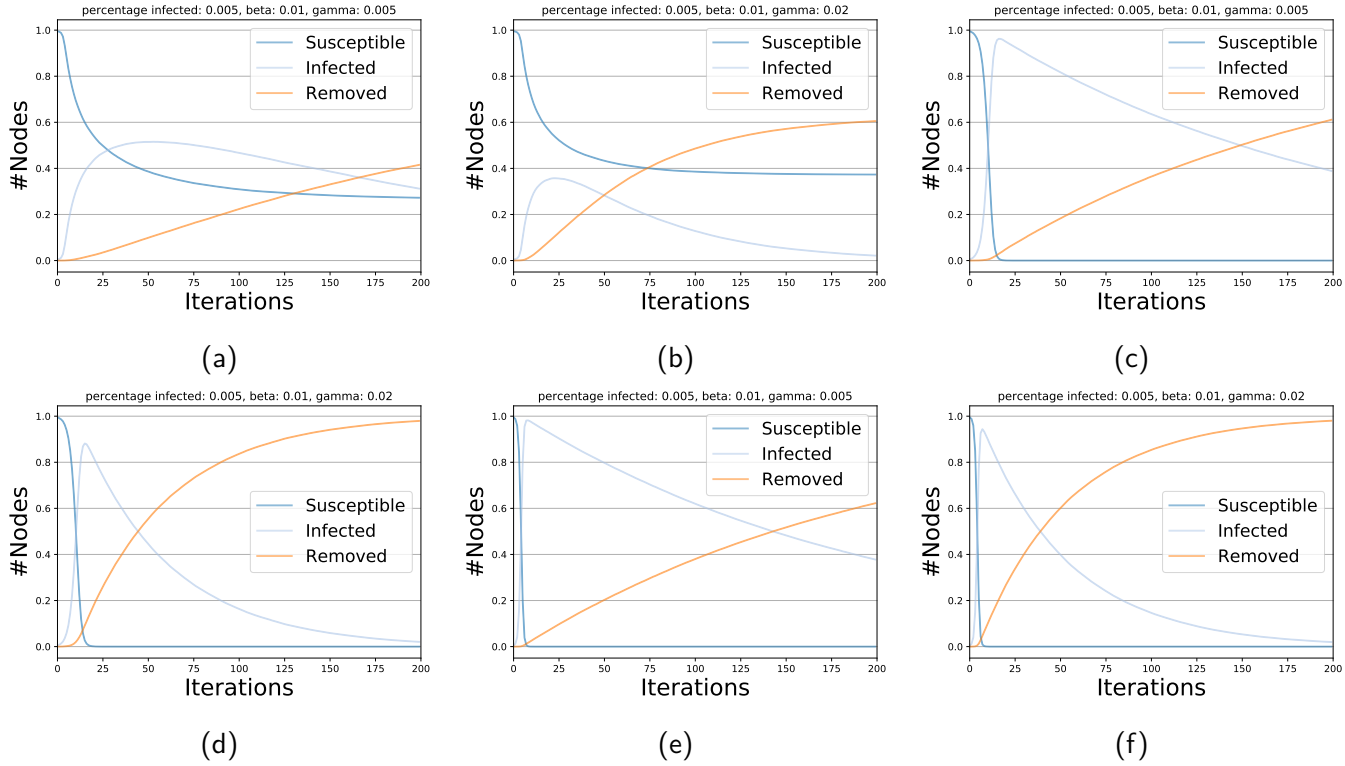


Figure 7.3: In Figure 7.3a and 7.3b we can see the representation of the diffusion on the original network both for the case in which γ is smaller than β and the other way around. The same kind of representation is plotted for the Erdős–Rényi network in Figure 7.3c and 7.3d and for the Barabási–Albert network in Figure 7.3e and 7.3f.

7.4 Threshold model

Finally we describe the application of the **Threshold model** both on the original network and the synthetic ones. In order to test this model we've choosen to apply a threshold τ eguals to 0.10, the diffusion of the infection for this model is represented in Figure 7.4. As we can see, for the original network we have that almost all the nodes become infected within the first 20 model's iterations, due to the fact that the value choosen for the threshold results to be sufficient for the spreading of the infection. If we change the threshold's value, this time using 0.20, we can observe that the original network become immune to the infection, thanks to its internal structure. We can observe the same immunity in the Erdős–Rényi and Barabási–Albert network for the original threshold's value.

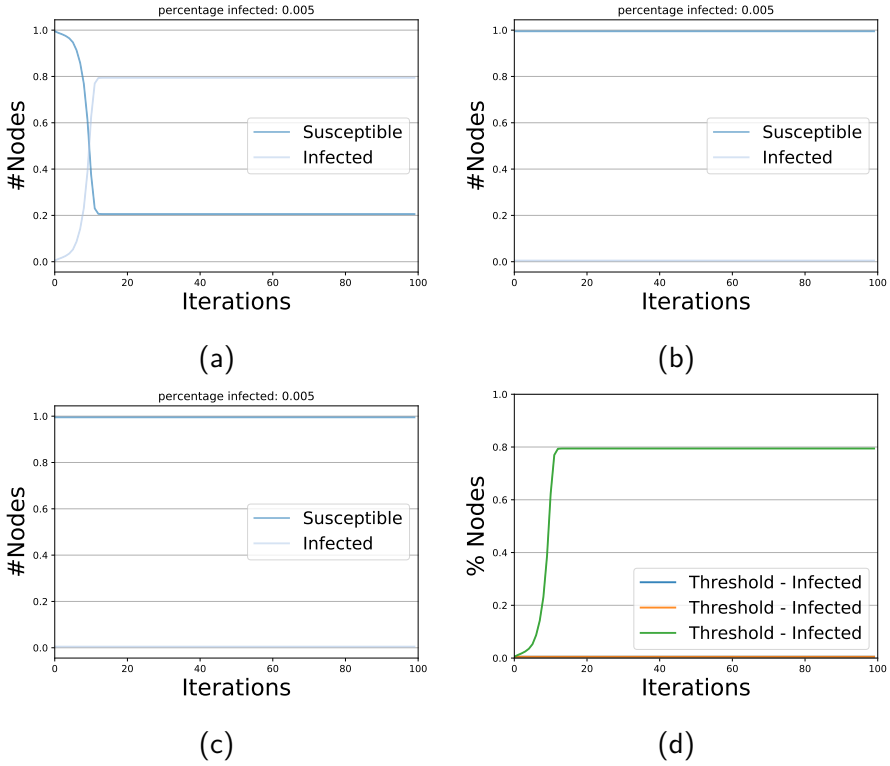


Figure 7.4: In Figure 7.4a is represented the diffusion of the infection for the original network, while in Figure 7.4b and 7.4c are represented the cases for the Erdős–Rényi and the Barabási–Albert network, respectively. A comparison between the three networks is represented in Figure 7.4d.

8 | Summary

References

[1] New York Times. *How Trump Consultants Exploited the Facebook Data of Millions*. https://www.nytimes.com/2018/03/17/us/politics/cambridge-analytica-trump-campaign.html. [Online; accessed 19-May-2018]. 2018.

[2] New York Times. *Cambridge Analytica and Facebook: The Scandal and the Fallout So Far*. https://www.nytimes.com/2018/04/04/us/politics/cambridge-analytica-scandal-fallout.html. [Online; accessed 19-May-2018]. 2018.

[3] Qi Ye, Bin Wu, and Bai Wang. "Distance Distribution and Average Shortest Path Length Estimation in Real-world Networks". In: *Proceedings of the 6th International Conference on Advanced Data Mining and Applications: Part I*. ADMA'10. Chongqing, China: Springer-Verlag, 2010, pp. 322–333. ISBN: 3-642-17315-2, 978-3-642-17315-8. URL: http://dl.acm.org/citation.cfm?id=1947599.1947633.

[4] Albert-László Barabási and Márton Pósfai. *Network Science*. http://networksciencebook.com/. 2016.

[5] Yang Jaewon and Leskovec Jure. *Defining and Evaluating Network Communities based on Ground-truth*. https://link.springer.com/article/10.1007/s10115-013-0693-z. 2015.

[6] Gergely Palla et al. *Uncovering the overlapping community structure of complex networks in nature and society*. http://dx.doi.org/10.1038/nature03607. 2005.

[7] G. Cordasco and L. Gargano. *Community detection via semi-synchronous label propagation algorithms*. https://ieeexplore.ieee.org/abstract/document/5730298/citations. 2010.

[8] Vincent D Blondel et al. *Fast unfolding of communities in large networks*. http://stacks.iop.org/1742-5468/2008/i=10/a=P10008. 2008.

[9] M. Girvan and M. E. J. Newman. *Community structure in social and biological networks*. http://www.pnas.org/content/99/12/7821. 2002.

[10] Michele Coscia et al. *DEMON: a Local-First Discovery Method for Overlapping Communities*. http://arxiv.org/abs/1206.0629. 2012.

[11] Giulio Rossetti, Luca Pappalardo, and Salvatore Rinzivillo. *A Novel Approach to Evaluate Community Detection Algorithms on Ground Truth*. https://doi.org/10.1007/978-3-319-30569-1_10. 2016.