LinkedIn's Companies Network

Tommaso Cavalieri
t.cavalieri@studenti.unipi.it
Student ID: 597707

Andrea Fedele
a.fedele7@studenti.unipi.it
Student ID: 599312

Federica Guiducci
f.guiducci2@studenti.unipi.it
Student ID: 600310

Valentina Olivotto
v.olivotto@studenti.unipi.it
Student ID: 600009

ABSTRACT

Is it feasible to infer how information is spread and shared between companies on the basis of the employees they have exchange in the past? The possibility to answer such a thoughtful question was explored by building and then analysing a network between companies with some standard social network analysis techniques[1].

KEYWORDS

Social Network Analysis, LinkedIn, Job's macro-sectors

ACM Reference Format:
Tommaso Cavalieri, Andrea Fedele, Federica Guiducci, and Valentina Olivotto. 2020. LinkedIn's Companies Network. In *Social Network Analysis '20.*

1 INTRODUCTION

The idea behind the analysis presented in this report was to explore all the characteristics of a network and compare it with some standard models such as the Erdös–Rényi, Watts–Strogatz or Barabási-Albert. The network taken into consideration was built from data that was retrieved from LinkedIn specifically for this purpose. Such data was exploited in order to connect various companies, which were represented as the nodes in the graph, through links that would represent the turnover of employees between them. Please note that a link connecting two companies A and B did not imply that a certain individual moved directly from company A to company B at some point in time. In fact, the building of such links was based on historical information crawled from LinkedIn on the employment of the selected sample of people, but without considering the chronological order, hence resulting in undirected links. Different approaches in the network building phase could have been possible, for example by constructing edges only on subsequent positions and thus resulting in directed links, but the reasoning behind the choice made for this study was to include in the analysis the whole career of the employees, especially given the high turnover rate in today's job market. Furthermore it seemed more interesting trying to understand whether a certain sector would show higher turnover than another than observing the temporal aspect of these individual's employment. A link between two different companies was then built if at least 2 persons had been employed in both companies at some point. Obviously the built network presented weighted links, with weights equal to the number of people that had worked in both the companies. Moreover, each link was also given an attribute based on the data retrieved from LinkedIn: the industry filter. Such feature was equal to the mode of the industry filters of the people that the two companies connected by that link had exchanged in the past. The analysis conducted on such network will be presented step by step in the following paragraphs.

2 DATA COLLECTION

Once established the purpose of the analysis, the first step required to fulfill the desired goal was to select an appropriate method for obtaining the data necessary to build the network as thought in the planning phase. In fact, there were no datasets already available online that could have been helpful for such purpose and it was therefore decided to crawl data from an online data source, i.e. a social network.

[1] **Project Repositories**
Data Collection:
https://github.com/sna-unipi/data-collection-2020_fedele_cavalieri_olivotto_guiducci
Analytical Tasks:
https://github.com/sna-unipi/network-analysis-analytical-tasks-2020_fedele_cavalieri_olivotto_guiducci
Report:
https://github.com/sna-unipi/project-report-2020_fedele_cavalieri_olivotto_guiducci

SNA '20, 2019/20, University of Pisa, Italy

Selected Data Sources

The most suitable social network for the described purpose seemed to be **LinkedIn**, since it is the one where the information regarding the job history of the users is more detailed and accurate. Furthermore, by exploiting the one-month free period offered by the platform, each of the authors created its own LinkedIn Premium account, in order to use its additional functionalities. One of them is the *Sales Navigator* platform: a tool that enables an advanced and targeted search for both single users and companies, which was vastly exploited for the data collection. Moreover such accounts guaranteed the possibility of exploring an unlimited number of profiles. The first task was to collect a list of urls of LinkedIn users, whose selection was restricted to people based in Italy belonging to different sectors in companies that went from 50 to more than 10.000 employees. The selected profiles were thus the sample on which the network construction was based and for each one of them the following information were properly saved: *name, current location, industrial sector of the company they were working for, the 5 latest job positions occupied along their location, education title, field of study* and *duration*. After downloading and properly saving the described data, the edges were identified by considering every individual's career as a set: the links were the intersections between such sets (please note that auto-links were not allowed, since they would have not been significant for the purpose of the analysis) while the nodes were the companies that had at least one edge connected to them.

Crawling Methodology and Assumptions

Since all the data used for the study was extracted from LinkedIn, the technique adopted to implement such idea was **web scraping**. The behaviour of a user was simulated, using the *Sales Navigator*'s search function to extract the urls of the profiles. Such search was restricted by applying some filters such as sector (e.g. *investment banking, insurances, logistics and supply chain* etc.), location (Italy) and the dimension of the company they were working for, measured in terms of employees (50-100, 100-200, 200-500, ...). Companies with less than 50 employees were excluded because their turnover was expected to be much lower. The explored sectors were selected on the base of the expected turnover in them, i.e. consultancy, IT and financial services. By varying the sector and dimension filters the crawled data was finally composed by a sample of 221,782 employees belonging to 23 different sectors. Once the list of urls was obtained, another web scraping method was implemented, again simulating the behaviour of a user, but this time visiting the profiles linked to the urls and extracting the aforementioned information from them. The tools used for the implementation were Python with its **Selenium** package and **Chrome Driver**.

The obtained dataset was then exploited to build the network as previously described. The results were not very satisfactory at first, since it was clear that some major data cleaning was needed. In fact, there were a lot of cases of misspelling or different names representing the same company which Python was not capable of detecting (e.g. *Accenture SPA* and *Accenture S.P.A.* were originally seen as different companies). Such cleaning was therefore implemented manually by looking at the list of links originally created. Once completed, the final network was composed of 14,875 nodes and 43,932 edges, instead of the 19,329 nodes and 47,874 edges initially found; it can be observed that it is a sparse graph, as in most of the real networks. Table 1 reports some interesting attributes of the built graph, from which it can be observed that the average degree is included between 1 and *ln(N)*: the network is thus **supercritical**, as expected from real networks, which should lead to the discovery of a giant component. Further analysis will be presented in the following sections.

Number of Nodes	14,875
Number of Edges	43,932
LMAX	110,625,375
AVG Degree	5.906

Table 1: Network's statistics

3 NETWORK CHARACTERIZATION

In order to give a deeper perspective into how the network built from the crawled data was composed, it was compared to other four synthetic models, whose parameters were set in such a way that they would have the exact same number of nodes of our graph and a similar amount of edges. Such models were the following:

- **Erdös–Rényi** with p = 0.00039 and undirected links;
- **Watts-Strogatz** with K = 6 and p = 0.1;
- **Barabási-Albert** with m = 3;
- **Configuration model** with the same degree distribution as BA.

Degree distribution analysis

By looking at the degree distribution in Figure 1 it is possible to assert that the company network was quite similar to the Barabási-Albert model (and so to the configuration model too), while it appeared very different from the Erdös–Rényi and the Watts-Strogatz. The observed graph followed a **power law distribution**, as could be expected from a real network. In fact, since alpha is included between 2 and 3, it is possible to collocate such graph in the **ultra-small world** in a **scale-free regime**. The presence of hubs in the network was also observed: indeed the creation of

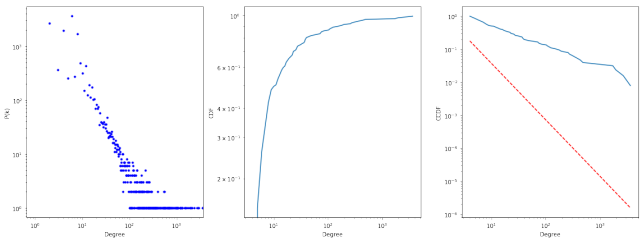


Figure 1: Degree distribution

links between nodes follows a preferential attachment to such hubs, which will be analysed deeply later in the report.

Connected components analysis

The analysed graph was composed by a total of 717 connected components, between which emerged a giant one composed of 12,693 nodes (85.33% of the whole graph), while each of the others included at most 12 nodes. Such composition can be appreciated in Figure 2. Given the graph semantic it could have been expected to obtain a high number of connected components, but instead a giant one was obtained with a surprisingly high cardinality of nodes. In the implemented Watts-Strogatz, Barabási-Albert and Configuration models a single connected component was obtained, whereas for what concerns the Erdös–Rényi model, 56 connected components were discovered: a giant one composed of 14,819 nodes (99.62% of the whole graph), while 54 of the others were formed by a single node and the last one by two nodes.

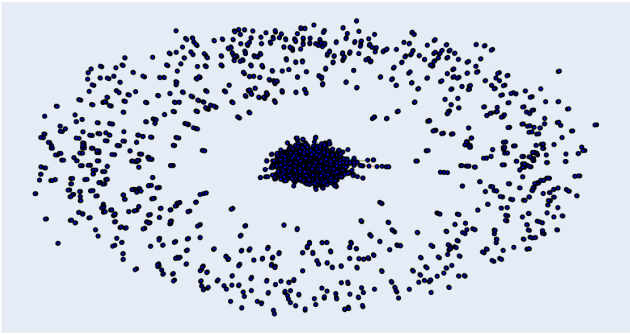


Figure 2: LinkedIn's Companies Network

Path analysis

For each giant connected component (both in the company graph and in the synthetic ones) a couple of distance measures were calculated such as the **diameter** and the **average shortest path**. Referring to the former, the company graph had a diameter equal to 11, same as the Erdös–Rényi's, while it resulted to be 16 for the Watts-Strogatz and 7 for the Barabási-Albert. Nonetheless the average shortest path resulted in our network was equal to 3.9784, being smaller than all the ones of the synthetic graphs as shown in Table 2. Such result was not surprising as for real-world networks the path length is usually $O(logn)$. The average shortest path increased up to 9.0782 when taking into consideration the weights of those links; such result was in line with the fact that the total sum of the weights is almost 225% of the total number of edges in the graph.

Network	Degree distribution	Path Lenght	Cluste-ring coeff.	Conn. Compo-nents
LinkedIn Network	Power law	3.9784 (weighted - 9.0782)	0.5970	717
ER	Poissonian	5.6742	0.0003	56
BA	Power law	4.4109	0.0039	1
WS	Poissonian	9.0362	0.4425	1
CM	Power law	4.4109	-	1

Table 2: Network's distances

Clustering coefficient - Density analysis

The graph's global clustering coefficient was equal to 0.6, which was way higher than the ones obtained in the synthetic graphs, even though the Watts-Strogatz model reached the value of 0.44. Moreover, by observing the ratio between local clustering coefficient and nodes degree it was highlighted how nodes with a higher degree presented a very low clustering coefficient, while low degree nodes varied way more, assuming clustering coefficient values throughout the whole range of its co-domain as shown in Figure 3. The observed density was very low (0.00039), as expected given the low number of links (43,932) in respect to the potential ones which is $N(N-1)/2$ and therefore larger than one hundred millions.

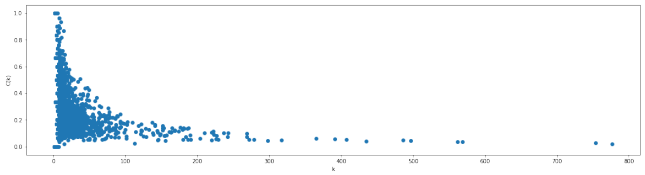


Figure 3: Local clustering coefficient vs Node degree

Centrality analysis

For this task different methods were implemented, each based on different definitions of *centrality*, i.e. **degree based**, **connectivity based** and **geometric based**. By looking at

the most central nodes obtained with the various measures, substantial differences could not be observed as the resulting central companies were always the same ones. In fact, only the order of such nodes changed from one method to the other, but the nodes themselves did not. Figure 4 shows the network's subgraph containing the 19 most central nodes, where the sectors are highlighted by the different link's colours. The two nodes that resulted to be the most central ones were *Accenture* and *Freelancer*, immediately followed by IT consultancy firms and banks. It could thus be hypothesized that between the sectors used for the data collection these two were the ones which presented the highest turnover in terms of employees. For what concerns the connectivity based methods, the *eigenvector centrality* and the *page rank* techniques were implemented. They both highlighted *Accenture* as the most central node, from which it could be deduced that a high number of people had moved from other important companies to this one or vice versa. Referring to the geometric based approaches, the only interesting results were obtained with the *betweenness*, even though the values of the first nodes were still very low; in this case the most central node was *Freelancer*, which could be both indicating autonomous workers or kind of an intermediary status that an employee assumes before moving from a firm to another. It was noticed that with such approach and with the connectivity based methods the centrality values of the nodes were decreasing, even though relatively low, while the *closeness centrality* and *harmonic centrality* were almost constant for many nodes (more than 300). This highlighted that such definitions of centrality were not as good for the analysed network, as the distance between a company and the rest of the network was not very useful to understand how employees move between them.

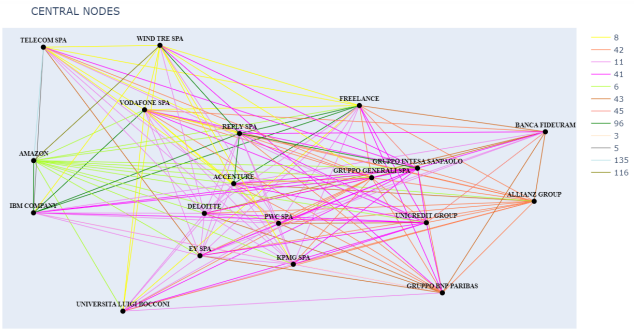


Figure 4: Central nodes

The last task implemented for the network characterization was about **assortativity mixing**: since the considered graph represents a professional network, it resulted to be neutral in respect to the degree assortativity measure. In particular the coefficient of Newman's assortativity was equal to -0.0269 and the associated degree correlation plot is shown in Figure 5.

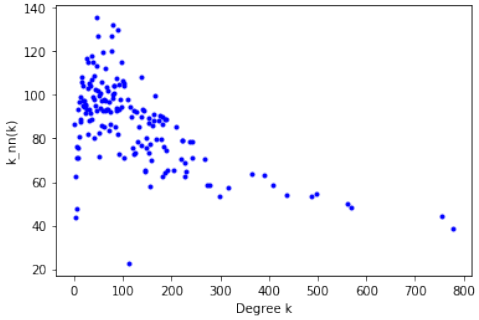


Figure 5: Degree assortativity correlation

4 TASK 1: COMMUNITY DISCOVERY

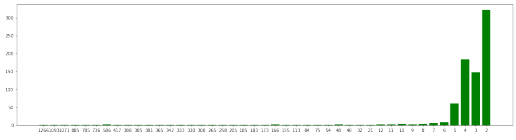
After characterizing the network as described in the previous paragraphs, more assignments were faced. The first one was community discovery, for which four different methods were selected and implemented: **K-clique**, **Louvain**, **Label Propagation** and **Demon-Angel**. The first step was the parameter tuning for such techniques, for which it was decided to use a random search in order to maximize the modularity measure. The parameters selected are reported in Table 3.

Model	Parameters
k - clique	*k = 3*
Louvain	*w = "weight"* *resolution = 0.8* *randomize = False*
Label	-
Demon	*epsilon =0.3* *min_com_size = 3*

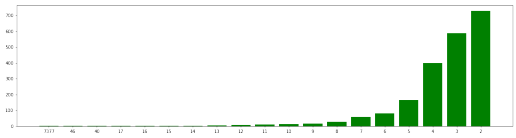
Table 3: Parameters

Internal evaluation

The results of the described implementations are shown in the Table 4, where it is possible to see that in all cases a large number of communities was obtained but with different cardinalities. Indeed, it was observed that Louvain and Demon methods discovered some small communities and others of medium size, while K-clique and Label Propagation found one big community and a lot of very small ones. This led us to believe that the Louvain and Demon approaches performed better since they splitted the network equally. Such cardinalities difference can be appreciated in Figure 6. As expected, Louvain and Label methods were able to find communities that were not overlapping and covering the whole



(a) Louvain



(b) Label propagation

Figure 6: Cardinality of the communities

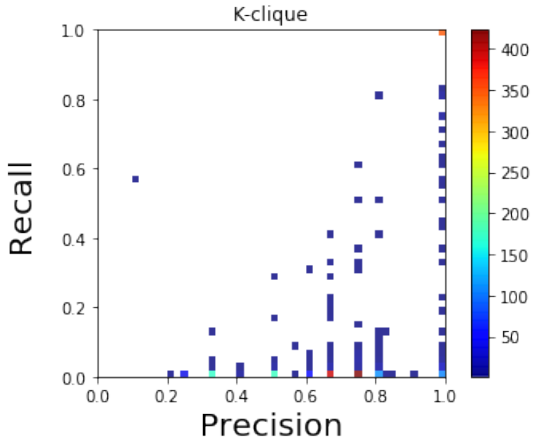
graph, thus creating a partition of the network, while K-clique and Demon created communities that did not include all the nodes of the original graph. In addition it could be observed that the average internal degree of Demon was higher than the others. Despite this, as far as we are concerned, Louvain algorithm was still the one that fitted better the analysed network and this was also confirmed by the fact that it reached the highest modularity score, as shown in Table 4.

	Num. of comm.	Overlapping	Coverage	Average Internal degree	Modularity
K-clique	2073	True	0.75	2.76	0.132
Louvain	785	False	1	2.05	0.624
Label	2110	False	1	1.97	0.244
Demon	1283	True	0.65	3.93	0.155

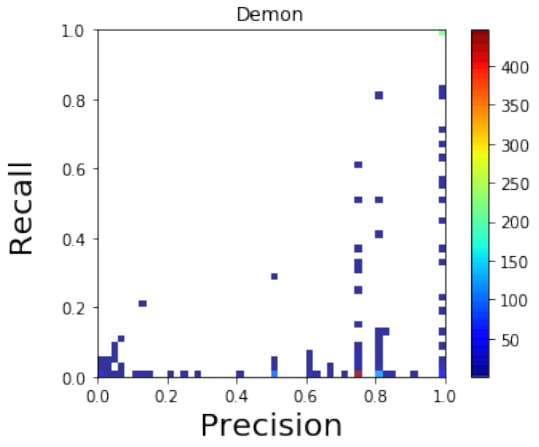
Table 4: Communities analysis

Comparison

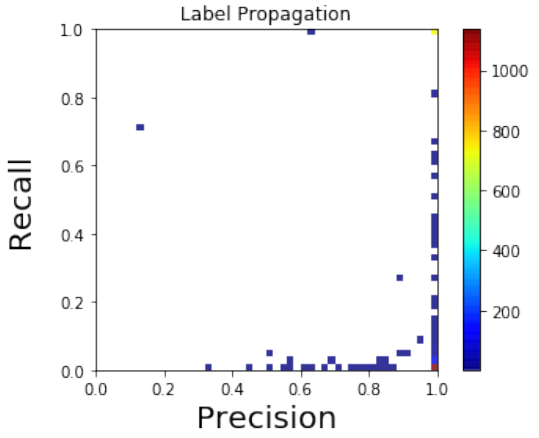
In order to compare the obtained communities the NMI and NF1 score were used. The **Normalized Mutual Information** can be used only under some coverage and overlapping constraints that were respected only by Louvain and Label propagation methods, which both reached a NMI equal to 0.61, highlighting how such partitions were not so different from one another. More interesting results were obtained thanks to the **NF1** score computation, in which it was decided to use as ground truth communities the ones resulting from the Louvain partition, since it had been identified as the best one during the internal evaluation. The results for each pair of methods were visualized in a precision-recall plot and the average F1 measure was calculated to better understand such comparisons. Plots about K-clique and Demon methods highlighted that there were a lot of pairings having



(a) K-clique



(b) Demon



(c) Label Propagation

Figure 7: NF1 analysis

low recall. Moreover, the overall F1-mean score was very low: 0.21 for K-clique and 0.23 for Demon. On the other hand, the comparison with Label Propagation returned a higher F1-mean value equal to 0.37. In Figure 7 a lot of pairings with low recall values and high precision can be observed, which can be interpreted as the fact that Louvain communities were fragmented into smaller ones. The communities found with Louvain algorithm were analyzed using an additional semantic information: the industry attribute. It was noticed that the majority of such communities was mainly composed by nodes connected by links of the same sector. The histogram in Figure 8 represents the number of communities (y axis) in which the most present sector reached a certain percentage (x axis).

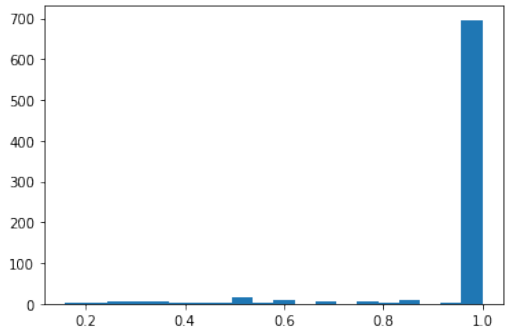


Figure 8: Communities industry composition

Figure 9 shows a clear example where a community was composed by 82% of links having the *insurance sector* as industry (orange edges). More analysis on the communities will be explored in Task 4.

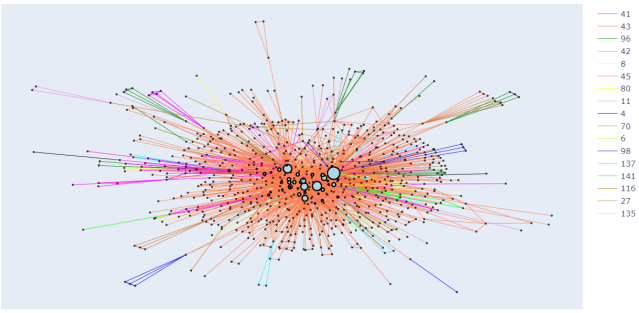


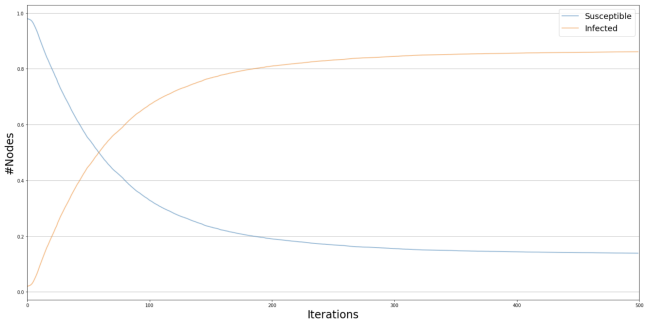
Figure 9: Community 4

5 TASK 2: SPREADING

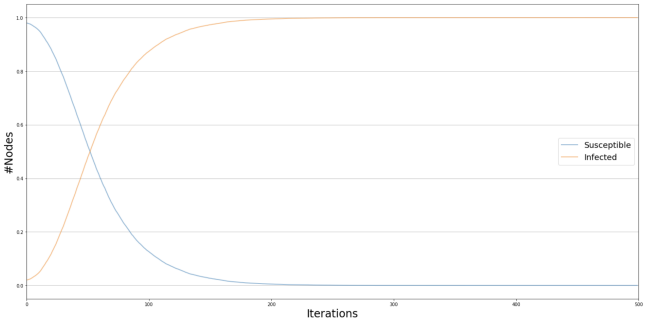
Another interesting problem that was analysed in this study was the diffusion: the decision was to implement some spreading techniques to compare the results obtained on the company graph with the ones on the synthetic models. Unfortunately, the interpretation of such techniques was not easy, since the analysed graph presented companies as nodes instead of people. Therefore, it was not possible to think of the spreading as usual, i.e. as an epidemics that infects people through their connections, but instead it should be seen as how easily an information or an idea might spread in a network composed as the one investigated in this study. The implemented model was the basic **SI model** since the *recovered* status and the possibility of becoming *suscetible* after having been *infected* used respectively by the SIR and SIS models would have been meaningless in this analysis. For this reason only the results obtained with the SI model will be presented in this report. Due to the interpretability problems just exposed, the parameters tuning was also hard to face: usually the parameters are estimations calculated on the basis of the observed data, which are not available in this case; hence a random search was implemented between different values for the parameters and the best results were obtained for *beta*=0.01 and with an initial percentage of infected equal to 2% of the 'population'. After running 500 iterations, it could be appreciated how the portion of infected became larger than the one of the susceptible already after 60/70 iterations and afterwards it stabilized around 0.85 as shown in Figure 10. The synthetic model that had the most similar behaviour was the Barabási-Albert, as shown also by the similarities between deltas in Figure 11. In fact, even though the Barabási-Albert, Erdös–Rényi and Watts-Strogatz models all allowed to reach a 100% infected status in the given graph, the Barabási-Albert is the one that seemed to reach the 50-50 situation in the same amount of iterations as the company graph.

Another implemented technique was the **Profile model**[2], introduced in 2017 by Milli et al. Such model assumes that the diffusion process is only apparent: each node decides whether to adopt or not a given behavior, once known its existence, only on the basis of its own interests. This model seemed to be the most appropriate for the purpose of this study, since the goal was not to analyse the actual spreading of an epidemic. In this model the diffusion process starts from a set of nodes that have already adopted a given behaviour S, which was chosen in three different ways: once at random, once selecting the nodes that emerged as central from the centrality analysis and once selecting the nodes not belonging to the giant connected component. For each of the susceptible nodes in the neighborhood of a node u in S, an unbalanced coin is flipped, where the unbalance is given by the personal profile of the susceptible node, equal to 15% for every node in our implementation, and in case of positive result the susceptible node will adopt the behaviour, thus becoming infected. With this model it was also possible to

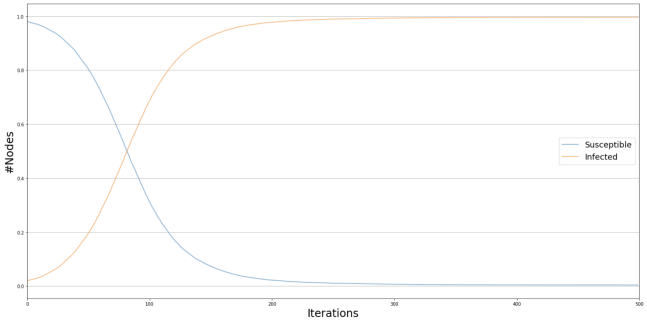
[2]https://ndlib.readthedocs.io/en/latest/reference/models/epidemics/Profile.html



(a) Company graph

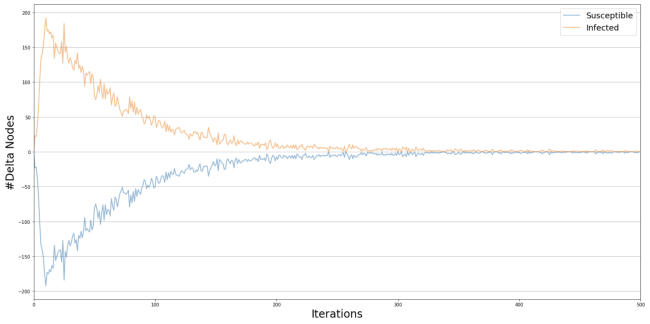


(b) Barabási-Albert

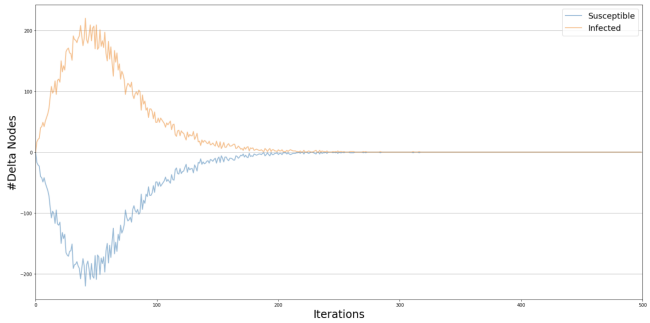


(c) Erdös–Rényi

Figure 10: SI diffusion comparison



(a) Company graph



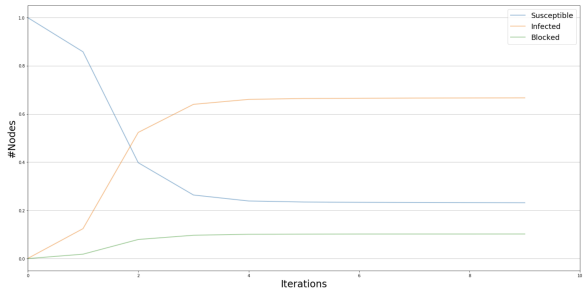
(b) Barabási-Albert

Figure 11: Delta nodes comparison

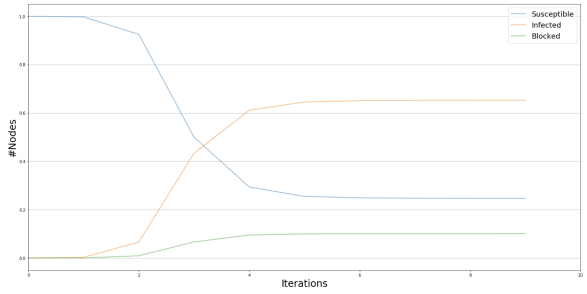
define the blocked status: after having rejected the adoption with a probability set equal to 10%, a node becomes immune to the infection. Moreover, at every iteration the 0.1% of the nodes spontaneously became infected due to endogenous effects. With the described configuration, a lot less iterations were sufficient (just 10) to appreciate the trend of the diffusion. The standard models behaved very similarly between each other with this method, but none of them seem to be particularly close to the behaviour of the company graph, which is presented in Figure 12. Subfigure 12a represents the diffusion when the status *infected* was assigned to the central nodes identified in the previous sections, subfigure 12b shows the results obtained by applying the same model, but randomly selecting the initially infected nodes, while in subfigure 12c are plotted the results obtained by selecting the ones not belonging to the giant component. It can be appreciated how, as could be expected, the increase in the infected and the decrease in the susceptible is slower when starting from random nodes, while it reaches its peak already after 3 iterations when nodes as *Accenture*, *Deloitte* or *Gruppo Generali SPA* are infected at time 0. The third implemented initialization the diffusion went as expected too, slowing down the diffusion in respect to the other described cases.

6 TASK 3: LINK PREDICTION

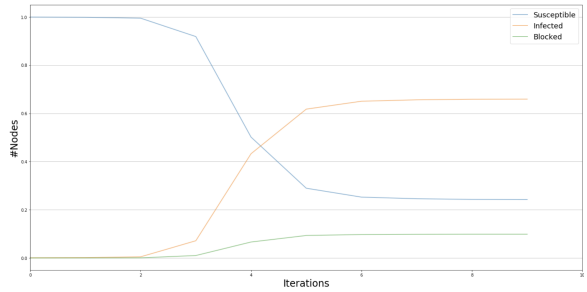
The third completed assignment had the goal of understanding how the studied network might evolve, by predicting the edges that could appear in it in the future. Hence, the idea behind this task was to estimate which companies will eventually exchange employees. Please note that the link prediction task could also be interpreted differently since it was applied to a network that was not given, but derived from the downloaded data. In fact, some links could be missing from the graph not because they were absent in general (no people moving between the two companies), but because there was no evidence of such movement in the retrieved data. For



(a) Central nodes infected



(b) Random initialization



(c) Marginal nodes infected

Figure 12: Profile model with different initializations

this reason link prediction might also be seen as the way of predicting the links that the data crawling methodology failed to capture. Unfortunately, this operation had a high computational complexity since the set of possible edges to be predicted is $O(|V|^2)$ where V is the number of nodes. For this reason, it was necessary to implement the selected techniques on a random subsample of the giant connected component, obtained as a subgraph of 1000 nodes. Such subgraph was then splitted into training and test sets with an 80-20 split in order to enable the evaluation of the performed procedures. Such link predictions methods were all unsupervised, specifically the **Adamic Adar**, **Common Neighbours**, **Jaccard**, **Katz** and **SimRank**, whose ROC curves are plotted in Figure 13. The associated Area Under the Curve (AUC) is presented in Table 5, where it is possible to appreciate how the predictors based on neighbourhood measures performed very similarly and they were all outperformed both by the Katz (10 times higher) and the SimRank (1000 times higher). Some further explorations were made, for example the intersection between the top-300 predicted links for each technique, but unfortunately they did not bring much additional information. Such intersection allowed to find 15 common links that had been given a positive score by all the implemented methods and for this reason such edges were analysed further. As could be expected, the nodes linked by these algorithms were mostly nodes belonging to the same sector. For example, one of the predicted links was *E-Work SPA - Ali SPA*, where both nodes were employment agencies, or *Software AG - Dassault Systemes*, where both company operate in the software sector. Despite this, it was not possible to find other remarkable results for this task. It was thus complicated to asses whether the incredible difference between the methods' performances was due to the incapability of some of them to adapt to the given data or it was just a peculiarity of the analysed network. Such conclusions were hard to be drawn also because it was not possible to implement the task on the whole network, but only to a portion of it that covered less than 10% of the nodes.

METHOD	AUC
ADAMIC ADAR	0.00038
COMMON NEIGHBOURS	0.00032
JACCARD	0.00028
KATZ	0.00338
SIMRANK	0.38377

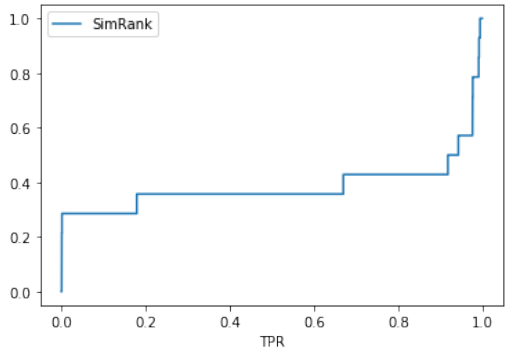
Table 5: Area Under the Curve

7 TASK 4: SECTOR ANALYSIS

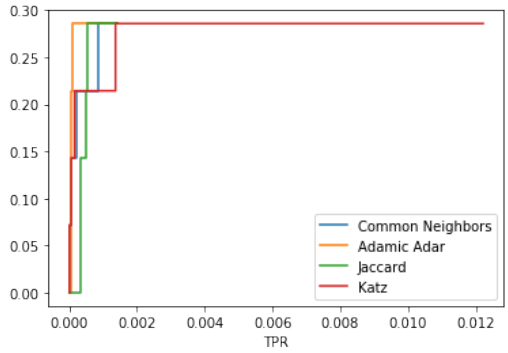
This task was thought to highlight an aspect of the crawled data that had been previously ignored: the links' industry sector. Such assignment was composed of two parts: the first one consisted in building a classifier that, given some features of the nodes that a link connected, was able to identify the industry sector associated to such link; the second part instead focused on the semantics of the graph by means of a deeper analysis on the distribution of such sectors.

Edge classification with Random Forest

This section describes how a dataset was derived from the network and then used in a machine learning classification problem. Each record of such dataset described a link between two nodes and the following features were extracted from the two nodes connected by it and averaged together: *Triangles*, *Local Clustering Coef ficient*, *Degree*, *Eigenvector*



(a) SimRank



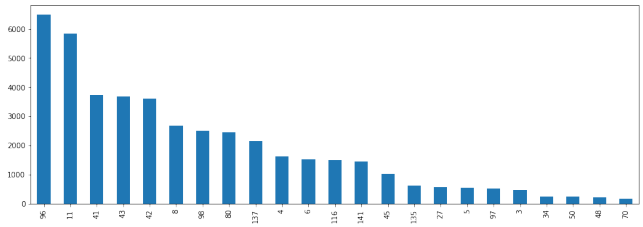
(b) Other techniques

Figure 13: ROC curves

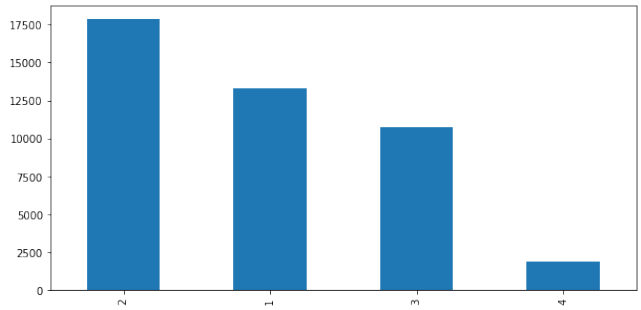
CONSULTING (2)	OTHER (4)
11 - Managerial consulting	34 - Food and beverage
41 - Banking sector	27 - Retail
42 - Insurances	48 - Building
43 - Financial services	50 - Architecture and planning
45 - Investment banking	135 - Mechanical and industrial engineering
PR (3)	**INFORMATICS (1)**
80 - Marketing and advertising	3 - Hardware
137 - Human resources	4 - Software
98 - Public relations and communications	5 - Computer networks
97 - Market researches	6 - Internet
70 - Research	8 - Telecommunications
116 - Logistics and supply chain	96 - Informatics and services
141 - International trading and developement	

Table 6: Macro-sectors composition

Centrality, *PageRange Centrality*, *Closeness Centrality*, *Harmonic Centrality* and *Betweeness Centrality*. The information about the nodes' name and the link weight were also maintained. Starting from the Louvian communities, the community number of the nodes connected by the links were saved too and then swapped in a sorting like fashion in order to preserve the graph undirectedness. The two values were then joined in a unique feature, to which One Hot Encoding was later applied. The selected target feature was *Industry*, which aimed to classify the industry sector of a link considering the attributes of the two nodes it connected. The industry feature was composed of 23 unique values and thus the analysed problem was a multi-classification one. Such attribute was later aggregated into 4 macro-sectors, as described in Table 6. The resulting dataset was composed of 43,919 rows and 1,374 columns. Figure 14 highlights the target class unbalanced distribution due to the data collection.



(a) Sectors



(b) Macro-sectors

Figure 14: Industry distribution

It was therefore established to use the Random Forest (RF) classifier, whose results will be described in terms of performance. It was then decided to apply the **LIME** and **LORE** algorithms in order to identify, explain and interpret some classification rules that could be extracted.

Random Forest

RF parameters were identified by means of a *Grid Search* based on the following parameters: ***criterion*** (*Gini*/*Entropy*) and ***class_weight*** (*balanced_subsample*/*balanced*). The number of trees imposed on the classifier was identified by means of an accuracy trend plot based on the results of a single Decision Tree. The *Grid Search* results are shown in Table 7, while Figure 15 shows the correlation between the number of estimators and the model accuracy (only for macro-sectors). In both cases it was clear that, after a certain threshold, the level of accuracy remained stable and the number of estimators was hence set around 200 to avoid overfitting.

Random Forest for industry sectors	*n_estimators = 209, class_weight = 'balanced', criterion = "Gini"*
Random Forest for macro-sectors	*n_estimators = 225, class_weight = 'balanced', criterion = 'entropy'*

Table 7: Random Forest GridSearch

The results of the RF based on the sectors are reported in the Figure 16. Unfortunately, the results of such classification were not optimal in terms of the considered evaluation measures; poor performances could be due to the fact that 9 sectors out of 23 contained a very low number of records and therefore the RF may not have been able to correctly capture their characteristics. However, taking into account the fact that it was a multi-label classification and that the number of labels to be classified was high (23), the results could still be considered satisfactory.

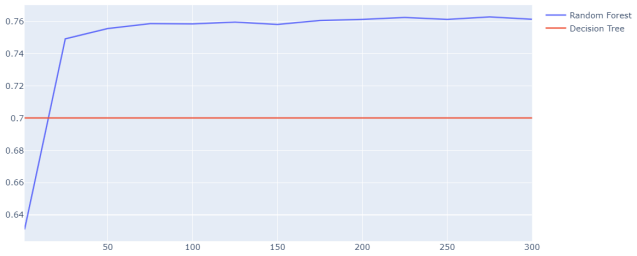


Figure 15: RF accuracy trend compared with DT accuracy

It is possible to notice an improvement in terms of performance when aggregating sectors into macro-sectors, as shown in Figure 17. From the Random Forest feature importance analysis, it was possible to notice that none of the features reached an importance higher than 8%. Another information that could be retrieved from the FI is that the most important features are those related to the nodes characteristics, immediately followed by the features that identified links connecting two nodes belonging to the same community. For these reasons it seemed necessary and interesting to make a deeper analysis on the communities and the sectors. The FI obtained from macro-sectors analysis are shown in the Figure 18.

```
Accuracy 0.5587431693989071
F1-score [0.36283186 0.35142119 0.36078431 0.44308446 0.62027594 0.56749025
 0.5480226  0.47435897 0.55884645 0.7323818  0.41666667 0.35033259
 0.48062016 0.29457364 0.20454545 0.6113002  0.59041591 0.62135922
 0.63004172 0.60858896 0.3480663  0.55208333 0.54594595]
              precision    recall  f1-score   support

           3       0.48      0.29      0.36       140
           4       0.47      0.28      0.35       484
           5       0.48      0.29      0.36       160
           6       0.51      0.39      0.44       460
           8       0.60      0.64      0.62       804
          11       0.52      0.62      0.57      1755
          27       0.53      0.57      0.55       170
          34       0.46      0.49      0.47        76
          41       0.50      0.64      0.56      1122
          42       0.71      0.76      0.73      1084
          43       0.48      0.37      0.42      1101
          45       0.55      0.26      0.35       307
          48       0.50      0.46      0.48        67
          50       0.36      0.25      0.29        76
          70       0.26      0.17      0.20        53
          80       0.61      0.61      0.61       737
          96       0.53      0.67      0.59      1949
          97       0.64      0.60      0.62       159
          98       0.66      0.60      0.63       754
         116       0.68      0.55      0.61       450
         135       0.36      0.34      0.35       187
         137       0.63      0.49      0.55       647
         141       0.66      0.47      0.55       434

    accuracy                           0.56     13176
   macro avg       0.53      0.47      0.49     13176
weighted avg       0.56      0.56      0.55     13176
```

Figure 16: RF sectors results

```
Accuracy 0.7607771706132362
F1-score [0.74788872 0.82849365 0.69878376 0.49736565]
              precision    recall  f1-score   support

           1       0.74      0.75      0.75      3997
           2       0.81      0.85      0.83      5370
           3       0.71      0.68      0.70      3234
           4       0.63      0.41      0.50       575

    accuracy                           0.76     13176
   macro avg       0.72      0.67      0.69     13176
weighted avg       0.76      0.76      0.76     13176
```

Figure 17: RF macro-sectors results

Explainability

LIME[3] and **LORE**[4] local explanation techniques were used in this task in order to understand the logic behind the classification rules. 50 records were selected randomly and both techniques were applied to them and compared in order to analyse the classifier's decisions. LIME permitted to understand, for each record, the target class in terms of probability, the local feature importance, which attribute pushed towards a certain label and how much. LORE, on the other hand, allowed us to get a better insight on the classification rules

[3]Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. Why should i trust you?: Explaining the predictions of any classifier. KDD.
[4]Riccardo Guidotti, Anna Monreale, Salvatore Ruggieri, Dino Pedreschi, Franco Turini, and Fosca Giannotti. 2018. Local rule-based explanations of black box decision systems. arXiv preprint arXiv:1805.10820

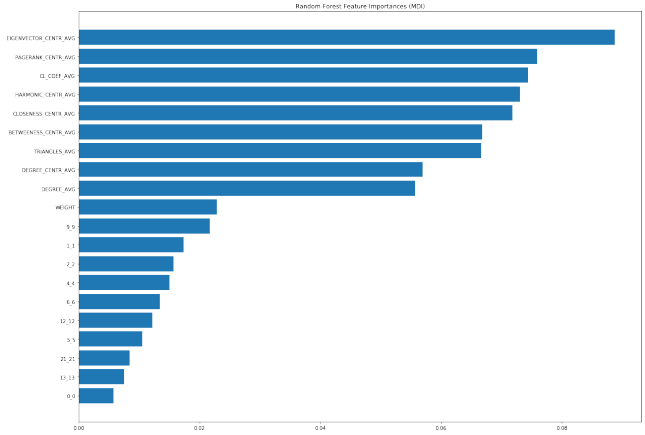


Figure 18: Feature importance of macro-sectors analysis

and the reason why the RF decided to classify a record with a certain class. It also gave the idea of what attributes should be changed in a record in order to obtain a different classification result. The selected records were carefully analyzed by class in order to find a common pattern of rules and local features importance. The LIME results showed a clear difference in terms of features local importance according to the target class, even tough they all had a small importance (from 0.01 to 0.08). In fact, *Betweeness* and *pagerank similarity* were the most important for identifying macro-sector 1, *Eigenvectors centrality* and *triangles number* were important for macro-sector 2 and *Eigenvectors* and *Harmonic Centrality* were important for macro-sector 3 and 4. The low value of the local importance did not help in getting any insight on the local feature importance. By analyzing the LORE results and comparing the rules and counter rules with the local feature importance of each record, it seemed clear that the community feature was the most important one. In fact, even if for some records the counter rule suggested to change various attributes, varying the community feature was enough to change the classification result. Analyzing further the rules it was clear that the RF classified the records basing its decision on the community feature that was set to 1 (=true), in particular it assigned to the record the sector that was most present in that community. Moreover, it was clear that the communities' feature where both nodes belonged to the same community were much more important than the others (as already shown in the Random Forest's feature importance). The analysis showed that the bigger communities were more important than the smaller ones since most of the model decision were based on them, which might be due to the unbalanced composition of the dataset. The feature with the highest value in terms of Random Forest importance was *1_1* and the LORE results confirmed it. In fact, for most of the examined records, setting this attribute to 1 was enough to change the classification result towards the class 1 (the most common macro-sector in community 1). Figure 19 shows an example of one of the records in the sample and what was its local feature importance towards the class 2 .

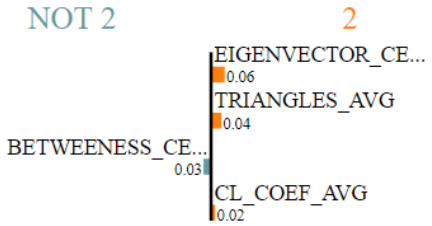


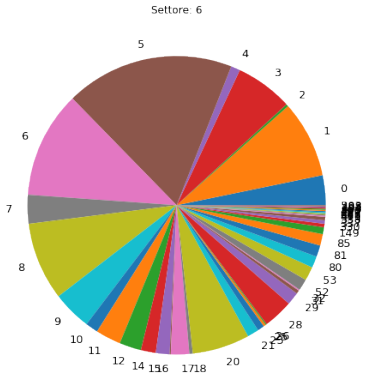
Figure 19: LIME record example

The results obtained by applying the LORE technique were pretty explanatory for this record. In fact the rules that brought the classifier to its decision are: *feature 1_1 <= 0.70* and *EIGENVECTOR_CENTR_AVG > 0.01*. The interesting thing to notice is that the counter rule changes the classification label from 2 to 1 by setting the current community 2_4 from 1 to 0 and the community 1_1 from 0 to 1. Finally, it can be observed that the described RF decided the classification label of a record according to the community to which the nodes connected by the link belong. Moreover, it set to the record the class of the most common industry sector in that community. It can be said that the community information was well kept and it turned out to be the most important one for the classification model despite the RF Feature Importance suggested differently.

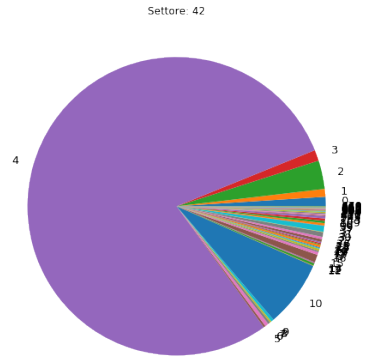
Sector characterization

This analysis was made to understand how the sectors influenced the structure of the graph studied and the community discovery. It was subdivided into three parts: the first one focused on exploring whether there was a relationship between sectors and communities, the second was to build a matrix to capture the correlation between sectors and the last one was to observe the sectors of the links connected to the central nodes.

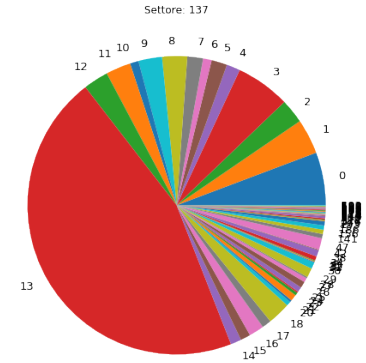
Sectors and communities. As previously highlighted in Task 2, the communities created were mainly composed by one single sector. The analysis was thus made by calculating, for each community, the percentage covered by each sector and, for each sector, its percentage in the different communities. After the computation of these values for all the different sectors, macro-sectors were also considered to better understand the results. Since the first communities were the biggest ones and they covered most of the network,



(a) Example of the IT macro-sector



(b) Example of the consulting macro-sector



(c) Example of the PR macro-sector

Figure 20: Communities distribution in different sectors

they were the only ones considered for this task. Specifically, even though an high percentage of IT sectors was discovered in communities 0, 1, 3 and 5, such sectors seemed to be uniformely distributed, being spread all over the communities. The sectors of the consulting macro-sector presented a different distribution: indeed the majority of these links were mostly found in the communities 0 and 2, apart from the *insurance sector*'s links that were in community 4. In addition, the analysis revealed also that vice versa in the community 4 was presented only the *insurance sector*, while in the communities 0 and 2 were present the other consulting sectors (41, 43 and 45). Differently from the IT macro-sector it was possible to notice that in this case sectors were concentrated in fewer communities. The difference was also consistent for the public relations macro-sector: in fact, all these sectors were present with high percentages in many different communities, revealing a weak connection between the sectors in it. Vice versa, thanks to the study of the communities, it was noticed that some medium communities were composed almost exclusively by the PR sectors (e.g. community 13 by the sector of *human resources*, community 19 by the sector of *logistics and supply chain*). The last macro-sector *Other* was composed by sectors of different fields and the results were trivial: such sectors are distributed over various communities. Figure 20 shows some examples of how a given sector distributed throughout the communities.

Sector A	Sector B	Common nodes
Managerial consulting	Informatics and services	546
Managerial consulting	Human resources	447
Financial services	Informatics and services	332
Informatics and services	Human resources	325
Managerial consulting	Telecommunications	320
Banking sector	Informatics and services	313
Managerial consulting	Software	306
Managerial consulting	Internet	305
Managerial consulting	Marketing and advertising	301
Financial services	Human resources	286

Table 8: Highly correlated sectors

Sectors correlation matrix. Given the fact that the available data was not in tabular form, but it was a network, it was impossible to build a proper correlation matrix, but it still seemed interesting to investigate the possible relationships between the 23 different sectors. The idea was to build a 23x23 symmetric matrix where each cell *(i,j)* stored the number of nodes that had both a link of the *sector i* and a link of the *sector j* attached to them. By doing so, the 253 obtained pairings all presented at least 9 nodes in common, with the highest value of 561 reached by *banking sector* and *financial services*. As could be expected, very high values were reached for couples of sectors belonging to the same macro-sector, as the one just mentioned; a list of the couples coming from different macro-sectors with the highest supports is presented in Table 6. It emerged that *managerial consulting* was the most present sector and the two most represented macro-sectors were Informatics and Consulting, reflecting their dominance in the analysed graph, as already

observed during the centrality analysis. Such result could be associated with the multidisciplinarity of the consultancy sector[5], capturing people with many different backgrounds and coming from other sectors. The described matrix is plotted in Figure 21.

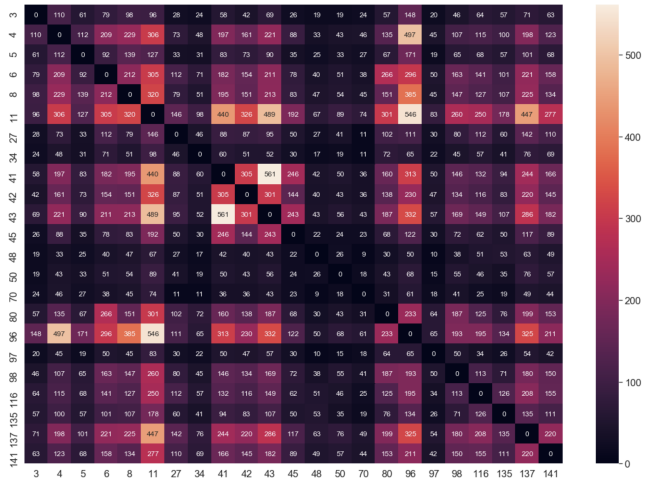


Figure 21: Correlation matrix

Central nodes' sectors. It was determined to further analyse the industry sectors distribution on each of the central nodes identified in the centrality analysis. In this section, different groups of nodes are described according to the macro-setor the belong to. Concerning the nodes belonging to the baking/financial sectors (*Allianz Group*, *Unicredit Group*, *Gruppo Intesa Sanpaolo*, *Gruppo BNP Paribas*, *Banca Fideuram*, *Gruppo Generali SPA*) there was a clear evidence that the sectors 41, 42, 43 and 45 were the most present, reaching a peak of 85% (*sector 42* in *Allianz Group*). In these nodes, sector 3, 6, 8, 11 and 96 were also present but with a percentage smaller than 10% separately. Analyzing the nodes belonging to the telecommunications sectors (*Vodafone SPA*, *Telecom SPA*, *Wind Tre SPA*) it was interesting to highlight that the most frequent sector in common was sector 8 (Telecommunications) with a percentage higher than 70%, reaching a peak of 86% on the *Wind Tre SPA* node. The minority percentage in these nodes came from sectors 5, 6, 11, 41, 43, 96 and 116 with a value smaller than 5% separately. Concerning the nodes belonging to the IT macro-sector (*Amazon*, and *IBM Company*), it was interesting to notice that their majority sector was not the same one. In fact, while 85% of links in *Amazon* came from sector 6 (Internet), 55% of the links in *IBM Company* came from sector 11 (Managerial Consulting).

[5]http://www.assoconsult.org/uploads/pages/attachments/39_assoconsult_osservatorio-2019-def-web.pdf

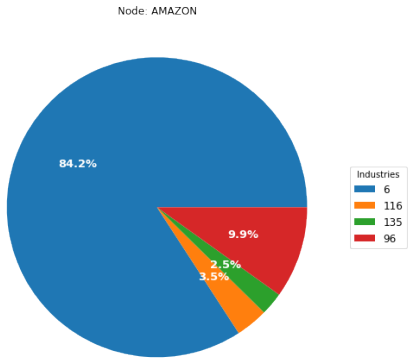


Figure 22: Amazon's sectors distribution

The sector distribution analysis on those nodes belonging to the consulting macro-sector (*Accenture*, *Deloitte*, *Reply SPA*, *EY SPA*, *PwC SPA* and *KPMG SPA*) showed that sector 11 was the most frequent in all of them: 60% in *Accenture* and 80+% in all of the others excluding *Reply SPA*. In fact, the most influencing sector in *Reply SPA* was 96 (Informatics and services) with 45% and the second most influencing was 11 with 30%. None of the minority sectors reached a percentage higher than 15% separately in any of these nodes.

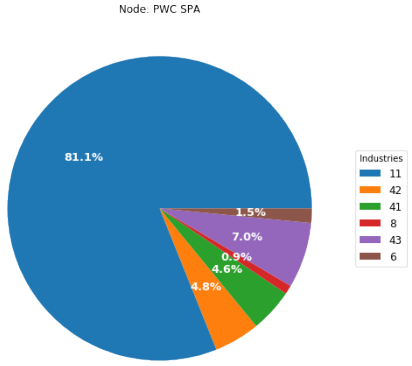


Figure 23: PwC SPA's sectors distribution

The only University node in the central ones was *Università Luigi Bocconi*, which was composed for a 70% by links belonging to sector 11 and the remaining minority ones (6, 8, 41, and 42) did not reach 10% separately. Interesting results were obtained for the node *Freelancer* which, following up the hypothesis done in the centrality task, did not actually represent a company, but more a temporary status (if not a life choice). The sector distribution found on this node is indeed the most sparse one, as shown in Figure 24.

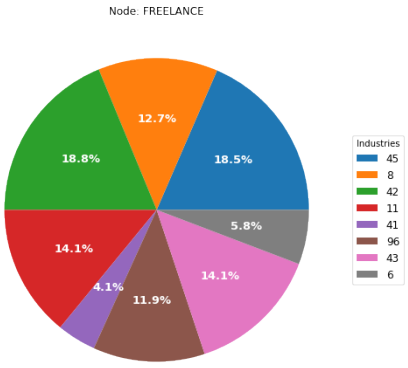


Figure 24: Freelancer's sectors distribution

8 DISCUSSION

The study described in this report was very time consuming, since just the data collection part alone required months of web scraping from several machines. Such problem was due to LinkedIn's policies for data crawling and at the same time it was impossible to find a valid alternative data source online to implement the project as thought during the planning phase. Moreover, once retrieved, the data needed a major cleaning, which was done only on the nodes but would have been even harder in case more attributes would have been required. Furthermore, some of the tasks such as the link prediction and path analysis had a very high computational complexity, which also contributed to the overall length of the project. Among the implemented tasks, network characterization and community discovery were the ones that gave the most satisfying results. It was in fact decided to make a deeper analysis on the discovered communities and their correlation with industry sectors in Task 4. The link prediction performed poorly and brought to extremely different results depending on the applied method, while diffusion did not present any problems in the implementation, but was still very hard to interpret due to the semantics of the network. The classification task implemented with the RF highlighted the importance of the nodes' community to identify the industry sector of the links connected to it. Finally, it was noticed that *consulting* and *IT* firms were the most present and also showed a lot of turnover between them. Such result could be expected because of the nature of such firms; in fact, consulting is a sector that shows an extremely high turnover when compared to the others and is characterized by multidisciplinarity, as previously mentioned.