Ministère de l'Enseignement Supérieur
et de la Recherche Scientifique

Université de Carthage

Ecole Polytechnique de Tunisie

وزارة التعليم العالي و البحث العلمي

جامعة قرطاج

المدرسة التونسية للتقنيات

Engineering Internship Report

Social Network Analysis

Biware Consulting

biware

From 17/06/2019 to 31/08/2019

Elaborated by: Fares FOURATI

Third year student at Tunisia Polytechnic School

Academic/University Year
2019-2020

Rue Elkhawarezmi BP 743 La Marsa 2078
Tel: 71 774 611 -- 71 774 699   Fax: 71 748 843
Site Web: www.ept.rnu.tn

نهج الخوارزمي ص.ب 743 المرسى 2078
الهاتف: 71 774 611 -- 71 774 699 الفاكس: 71 748 843
موقع الواب: www.ept.rnu.tn

Abstract

This work comes within the scope of the second year's Engineering Internship project at Biware Consulting. The project turns over Social Networks Analysis. Firstly, we have been through the theoretical aspects of graphs. Then, we have applied the social networks analysis methods on different social networks. We have analyzed a Facebook network and identified influencers and bridges in it. In addition, we have analyzed two bitcoin trading over-the-counter platforms and we have estimated the traders' trustworthiness in these two platforms.

Keywords: SNA (Social Networks Analysis), ML (Machine Learning), Python Programming, Facebook as a social network, detecting communities in social networks, detecting influencers in social networks, detecting bridges between communities in social networks, targeting classes in social networks, Trading Platforms, OTC (Over-the-Counter) trading, Trust as a necessity in trading networks, trader's trustworthiness estimation.

Acknowledgments

In conducting this report, I have received meaningful assistance from many quarters which we like to put on record here with deep gratitude and great pleasure.

First and foremost, I would like to express my sincere gratitude to the co-founder and CEO, Mr. Amine Boussarsar, for his support and encouragement, all along the internship.

I would also like to thank my supervisor, Ms. Roua Hammami, for her advices and remarks, all along the internship.

I would also like to thank all our teachers at Tunisia Polytechnic School for their continuous help and treasurable training during our study years.

Finally, special thanks go to the jury members who honored us by examining and evaluating this modest contribution.

Summary

List of Figures

Glossary Acronyms

- R&D: Research and Development
- SNA: Social Networks Analysis
- ML: Machine Learning
- OTC: Over the Counter
- BTC: Bitcoin

General Introduction

I have carried out my engineering internship in the R&D department (Biware Solutions) of Biware Consulting.

Biware Consulting is a Tunisian company created in 2011. It offers services related to the decision making. It has an R&D department called: Biware Solutions. The main purpose of this department is to develop products and solutions based on data science and artificial intelligence and all other emerging technologies.

As I have been through three major steps in my engineering project, I decided to divide the report on three major chapters.

The first chapter contains an introduction to some graph theory notions. Describing important notions, features, coefficients, and algorithms I have used in my analysis of different social networks.

The second chapter is mainly a basic application to the notions I have discovered during the first weeks of my internship. In fact, I applied what I have found in different papers turning over social network analysis and what I have learned from the courses I have enrolled. Although the first step is not highly complex, it helped me master graph notions and social network analysis methods which helped me go further in next applications.

Finally, the third chapter turns over the analysis of bitcoin trading network analysis. So, I have applied what I have learned during the bibliography period to solve the trust measurement in trading networks. In this chapter I describe an algorithm which was developed from scratch and implemented to estimate trustworthiness of each trader, the analysis of this algorithm, the study of its efficiency and convergence and the use of the estimated trustworthiness in the prediction of trust scores using machine learning algorithms.

Chapter 1

Social Networks Analysis Introduction

1. Introduction

In this first chapter, we will introduce some essential notions to social networks analysis. In fact, we have taken most of the definitions and notions from the course 'Applied Social Network Analysis by Michigan University'. [9]. This chapter contains the important highlights of the cited course.

2. Social Network Analysis

'Social network analysis (SNA) is the process of investigating social structures through the use of networks and graph theory. It characterizes networked structures in terms of *nodes* (individual actors, people, or things within the network) and the *ties*, *edges*, or *links* (relationships or interactions) that connect them.' [21]

Social Network Analysis is now involving because of the development of the technology. In fact, the world is getting more and more connected which increased the quantity of data and the volume of networks (Facebook, LinkedIn, WeChat, YouTube, …). We are in front of a huge puzzle to decrypt. Social network analysis can be applied in many fields, it can be applied in chemistry to analyze connections between atoms in the matter. SNA can be applied for marketing purposes. In addition, it can be applied in politics, psychology, sociology and economy.

3. Network definition and vocabulary

Network (or Graph): A representation of connections among a set of items.

- Items are called nodes (or vertices)
- Connections are called edges (or link or ties)

Directed Graph: edges have direction

Undirected Graph: edges have no direction

Weighted network: a network where edges are assigned a (typically numerical) weight. Edges can have many labels or attributes other than weights. Nodes can also have attributes.

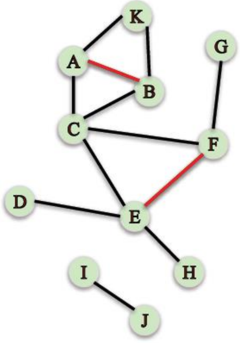
Signed network: a network where edges are assigned positive or negative sign.

Multigraph: A network where multiple edges can connect the same nodes (parallel edges).

Bipartite Graph: a graph whose nodes can be split into two sets *L* and *R* and every edge connects a node in *L* with a node in *R*.

4. Clustering coefficients

a. Local clustering coefficient of a node



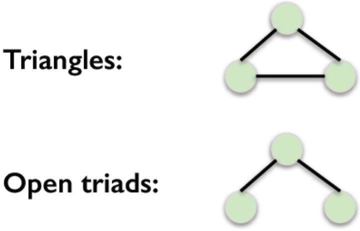
$$Clustering_{\text{coeff}} = \frac{\text{\# of pairs of C's friends who are friends}}{\text{\# of pairs of C's friends}}$$

b. Global clustering coefficient

First approach is to calculate the average of all the local clustering coefficients of all the nodes.

Second approach is to calculate the transitivity
Transitivity: Ratio of number of triangles and number of "open triads" in a network.

$$\text{Transitivity} = \frac{3 * \text{Number of closed triads}}{\text{Number of open triads}}$$



5. Distances in the graph

Distance between two nodes: length of the shortest path between them.

Eccentricity of a node *n* is the largest distance between *n* and all other nodes.

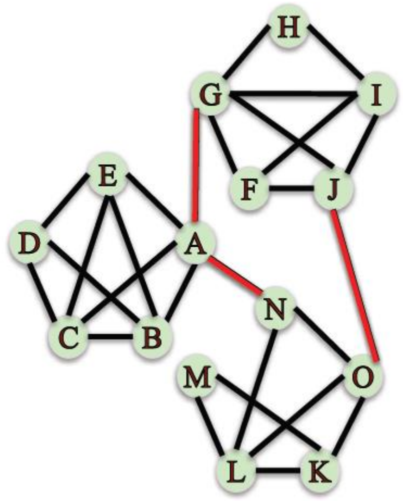
Average distance between every pair of nodes.

Diameter: maximum distance between any pair of nodes.

Radius: the minimum eccentricity in the graph.

Identifying central and peripheral nodes:
The **Periphery** is the set of nodes with eccentricity = diameter.
The **center** is the set of nodes with eccentricity = radius.

6. Connectedness of the graph



An undirected graph is **connected** if, for every pair node, there is a path between them.

However, if we remove edges A—G, A—N, and J—O, the graph becomes disconnected.

A directed graph is **weakly connected** if replacing all directed edges with undirected edges produces a connected undirected graph.

A directed graph is **Strongly connected** if for every pair node, there is a *directed* path between them.

Network robustness: the ability of a network to maintain its general structural properties when it faces failures or attacks.

Node connectivity: Minimum number of *nodes* needed to disconnect a graph or pair of nodes.

Edge connectivity: Minimum number of *edges* needed to disconnect a graph or pair of nodes.

7. Node Importance

a. Degree Centrality

We assume that important nodes have many connections.

The most basic measure of centrality: number of neighbors.

Undirected networks: use degree.

Directed networks: use in-degree or out-degree

$$C_{deg}(v) = \frac{d_v}{|N| - 1}$$

Where N number of nodes and d_v is the degree of the node *v*.

b. Closeness centrality

We assume that important nodes are close to other nodes.

$$C_{close}(v) = \frac{|N| - 1}{\sum_{u \in N \backslash \{v\}} d(v, u)}$$

Where N is the set of nodes in the network, and $d\ (v, u)$ =length of shortest path from v to u.

c. Betweenness centrality

We assume that important nodes connect other nodes.

$$C_{btw}(v) = \sum_{s,t \in N} \frac{\sigma_{s,t}\ (v)}{\sigma_{s,t}}$$

$\sigma_{s,t}$ is the number of shortest paths between nodes s and t.
$\sigma_{s,t}\ (v)$ is the number shortest paths between nodes s and t *that pass through node* v.

8. Popular algorithms

a. PageRank

Developed by Google founders to measure the importance of webpages from the hyperlink network structure. PageRank assigns a score of importance to each node.

Important nodes are those with many in-links from important pages. It can be used for any type of network, but it is mainly useful for directed networks.

The Algorithm:
1. Assign all nodes a PageRank of $1/n$
2. Perform the *Basic PageRank Update Rule k* times.

Basic PageRank Update Rule: Each node gives an equal share of its current PageRank to all the nodes it links to. The new PageRank of each node is the sum of all the PageRank it received from other nodes.

b. HITS Algorithm

Computing *k* iterations of the HITS algorithm to assign an *authority score* and *hub score* to each node.

1. Assign each node an authority and hub score of 1.
2. Apply the *Authority Update Rule:* each node's *authority* score is the sum of *hub* scores of each node that *points to it*.
3. Apply the *Hub Update Rule:* each node's hub score is the sum of authority scores of each node that *it points to*.
4. Normalize Authority and Hub scores
5. Repeat *k* times.

Chapter 2

Estimating people’s influence in a Facebook Social Network by analyzing people’s connections

1. Introduction

Identifying influencers is highly useful and has many applications. The spread of information is more efficient and faster when we target influencers in the network. For example, considering a company wanting to make an advertising, rather than sending messages to all the network (which is highly costly) it better detects the top influencers and bridges in the network and just targeting them and leaving them spread the information.

Identifying influencers can has serious impacts, like controlling groups, by manipulating the targeted influencer’s minds. For example, in the company itself, especially for big companies they can run algorithms on employee’s connections data to detect the biggest influencers in the company to control their influence on other employees.

In this part of the project we have used a Facebook social network. The Facebook social network we have used only contains edges(friendship) between nodes(persons). The only details we know about ‘x’ is the list of friends he has. We neither have details about the intensity connections nor about the frequency of contact between persons. So, detecting the influencers in the network was based only on the friendship connections. So, we used many classical social network analysis metrics and we have developed some to quantify the centrality, the popularity and the influence of each node. We have developed many functions (In Python) to calculate these metrics and to extract the top nodes for each metric.

2. Description of the data

The data we had was from Stanford SNAP Group [5]. The initial data is a table containing two columns one representing the person 1 and the second representing the person 2. So, each line describes a friendship between two persons.

	Node 1	Node 2
0	236	186
1	236	84
2	236	62
3	236	142
4	236	252

Figure 1: 5 lines of the initial data

In the network we have 84243 connection(edge) and 3959 person(node).

In the analysis, we will transform this initial table to graph G. We will represent each person with a node. Where U is a set of nodes. And we will represent the connection between nodes with an ed ge, where E is a set of edges. So, G=G (U, E).

As connections in this network are friendships so the graph is undirected. As we have no data other t han connections. The network is unweighted and unsigned. The problem is modeled with Undirected Unweighted Unsigned Social Network.

3. Finding top 10% for each metric

a. Degree Centrality

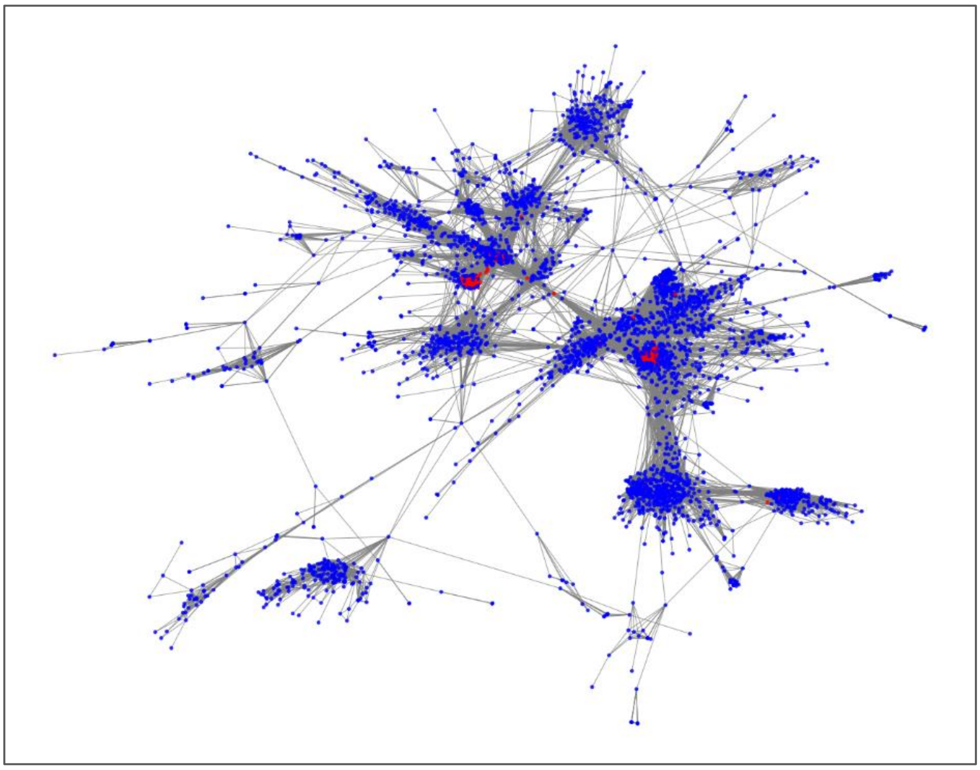


Figure 2: Red dots in the network have the highest degree centrality scores

(Their degree centralities belong to the highest 10% of the scores)

The chosen nodes are the top 10% nodes having the highest numbers of connections. Which is a good estimation of their influence in the network, but it is not enough to conclude.

b. Closeness Centrality

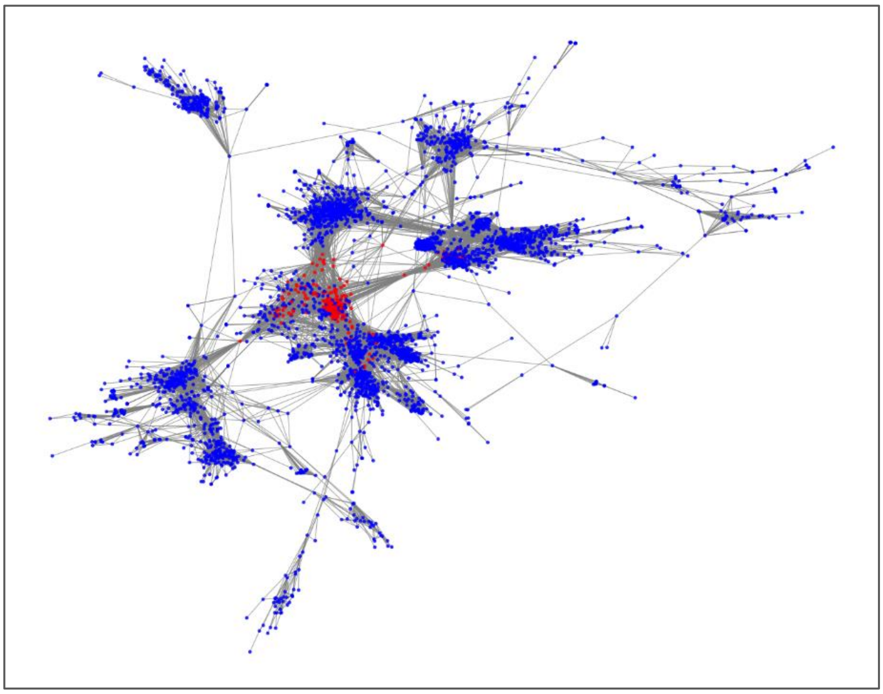


Figure 3: The red dots in the network have the highest closeness centrality scores

(Their closeness centralities belong to the highest 10% of the scores)

c. Betweenness Centrality

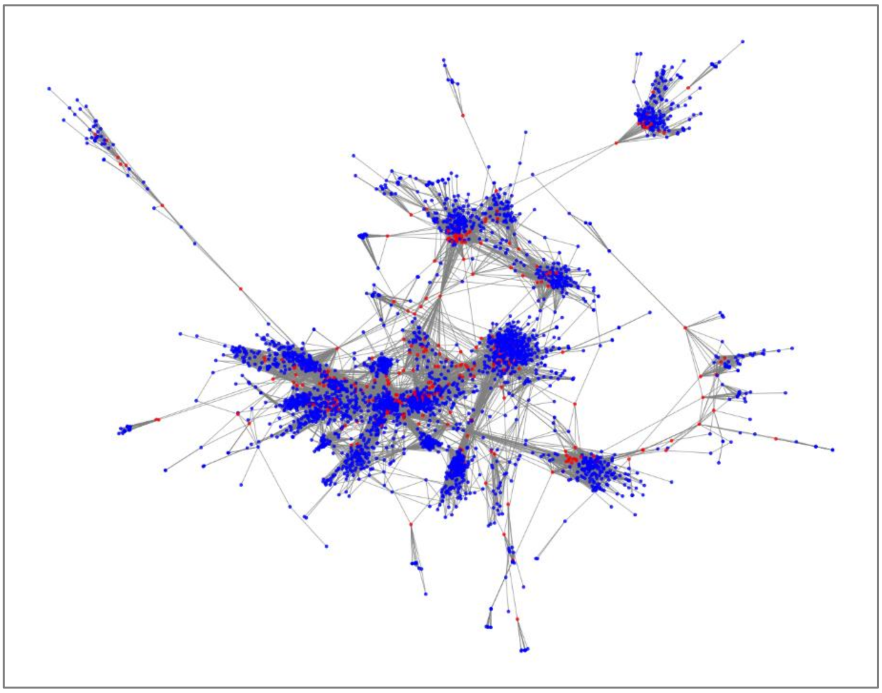


Figure 4: Red dots in the network have the highest betweenness centrality scores

4. Influencers and Bridges

a. Bridges

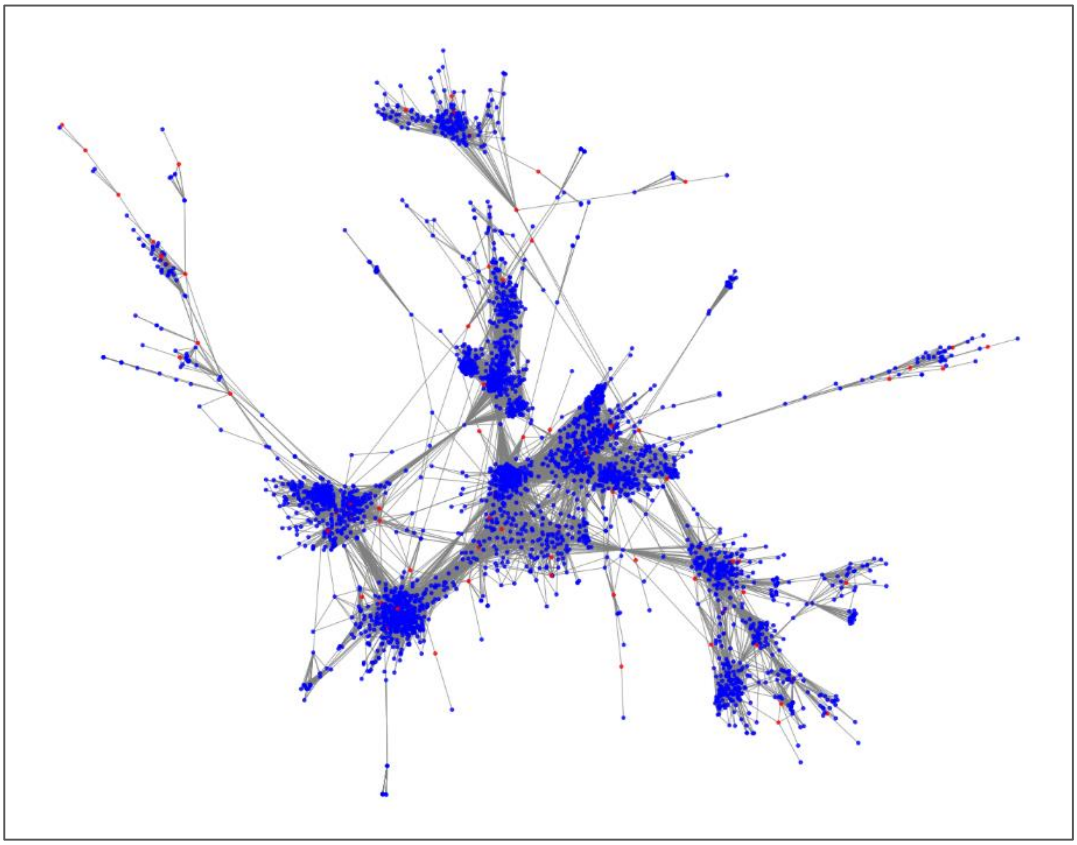


Figure 5: Red dots are the bridges in this Facebook social network

b. Influencers

We defined the influencers those who have higher degree centrality, closeness centrality and betweenness centrality. In other words, degree influence is the intersection of the three notions. Being influencer is being highly connected to the network, having a higher number of friends, being close to most of the network to have an impact on most of the network and being in between many optimal connections to have impact on these paths.

$$I \in (L1 \cap L2 \cap L3)$$

Where:

I: List of influencers

L1: List of nodes having highest n% degree centrality

L2: List of nodes having highest n% closeness centrality

L3: List of nodes having highest n% betweenness centrality ‘n%’ is defined by the number of influencers we want to extract from the network.

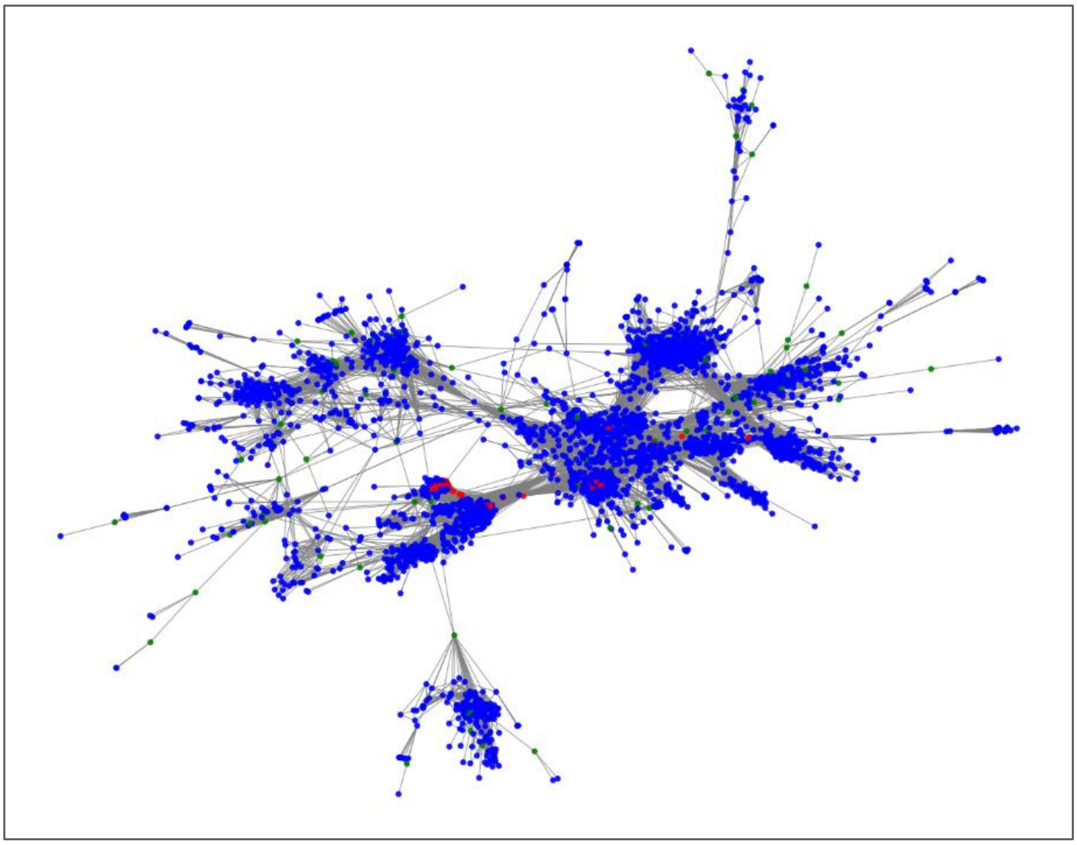


Figure 6: Red dots are the influencers and green dots are the bridges

c. Conclusion

It is possible to quantify many social notions, like popularity, influence, closeness and connectivity in social networks. As it is possible, to extract popular people in the network by just considering their connections on the network.

Chapter 3

Solving the Measurement of Trustworthiness Problem in Bitcoin Over-the-counter Platforms

1. Introduction

To accomplish any transaction, trust is highly required, because if we do not trust, we will not be able to judge the riskiness of the transaction. That is why it is essential to quantify the trustworthiness of the institution or the person we are dealing with. This paper shows an example where we use social network analysis to estimate the trustworthiness of individuals in some networks. The algorithm proposed in this paper can approach people's trustworthiness by considering many features in the network. In addition to solving the measurement of trust in trading platforms, this algorithm can be highly useful in other contexts, for example, to estimate people's skills (problem solving skills, social skills, technical skills, managerial skills, …) in social networks.

We have worked on the data of two trading platforms that are: bitcoin-OTC [10] and BTC-Alpha [11]. Previous literature has worked on these two datasets [1,6]. However, my work approached the topic differently: Both platforms represent a place to trade currencies and goods. Trading is achieved directly between counterparties without the intervention of the platform. As such, it is everyone's responsibility to act prudently and wisely and to choose whom to trust and whom to avoid. Traders should prevent fraudulent users if they are trading in these risky platforms. In this kind of trading, trust is a crucial element to proceed. As discussed by David and Andrew [2], Trust has three dimensions that are cognitive, emotional and behavioral. In fact, humans only need 100 ms of face exposure to others to extract the needed information to make a trustworthiness judgement [3]. However, in these kinds of online trading we cannot consider the emotional dimension of trust as we are not in real contact with the other traders. So, the remaining two involved dimensions in the experience are the cognitive and the behavioral dimensions. According to David and Andrew 'When faced by the totally unknown, we can gamble but we cannot trust. '[2]. Fortunately, these platforms offer some data on each user. In fact, the idea is that each user, can give a trust score to another one. So, each trader has received from some traders with whom he has made a transaction a trust score. Each trader has scored some of the other traders. The trust scores vary from +10 to -10. You can check the rating guidelines on bitcoin-OTC platform guide [4]. So, then the platform calculates the mean of the received scores for each one and rank people according to their mean received scores. In my point of view, this scoring and rating system is a good way to, at least, estimate the trustworthiness of the other traders. In this paper, we will discuss the possible frauds in Over-the-counter platforms. In addition, we will discuss the risks of frauds in this system. we will propose an algorithm that can simplify trader's life by estimating a trustworthiness score for each one and ranking people, rigorously and continuously, in the platform. we insist that my proposed score is different from just calculating the mean of the received trust scores, and we assume that the output of that algorithm is highly representative of the trustworthiness of the traders. We have used my algorithm's output to predict what traders have scored other traders, we have also measured the correlations between the estimated trustworthiness scores and other SNA features.

2. Risks in trading in Over-the-counter platforms

Varying degrees of counterparty risk exist in all financial transactions. Counterparty risk is a risk that both traders should consider when trading [14]. When you trade OTC, you engage in a transaction with anonymous traders. As explained in the bitcoin-OTC platform, 'You may send your bitcoin to the person, and never get anything back. Or you may send your $ currency to the person expecting bitcoin and get nothing in return.' [15]. That is why it is necessary to have the historic transactions of the trader or some equivalent data.

3. Critics of the rating system used in both platforms

To give traders a tool to classify people based on whom to trust and whom to avoid, the two platforms proposed a scoring system where each trader can score other traders depending on the intensity of the trust. So, people in the platform can access these scores and acquire some insight about whom they are trading with.

However, one of the major problems is that traders can create as much profiles as they can to score themselves with higher trust scores which can mislead people by making them think they are trustworthy. [15]

Another problem is that most of the traders have a maximum of two received trust scores which can lead to unrealistic trustworthiness score. (We found this when we analyzed both networks)

Furthermore, if we receive from the beginning, for a reason or another, a negative trustworthy score from a trader, we will be excluded because our mean received trustworthiness score will be negative. So, no one will take the courage to trade with someone having a negative score.

4. Modeling the Problem

As each trader scores some of the traders in the network by a given value, we have a weighted signed social network. We model the network by a graph G.

Let us note the set of nodes U, where each node represents a trader in the Bitcoin trading platform.

And let us note the set of edges E, where we represent the act of scoring by a directed edge labeled by the output of the weight function *w*: E → [-10,10], representing the score of trustworthiness given by a trustor node to a trustee node.

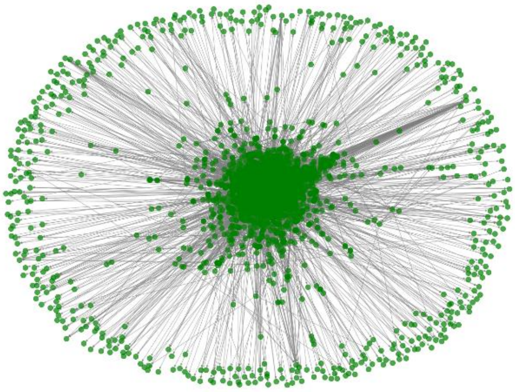


Figure 7 : BTC-Alpha network

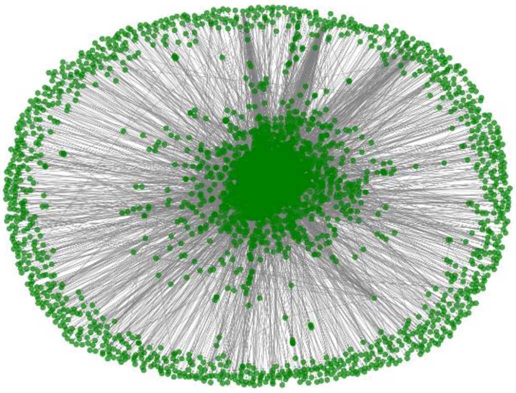


Figure 8 : Bitcoin-OTC Network

Both networks are weighted, signed and directed networks. The Bitcoin-OTC has 5881 nodes and 35592 edges. However, the BTC-Alpha network has 3783 nodes and 24186 edges.

5. Trustworthiness Algorithm

Definitions of the Hyperparameters:

α: the percentage taken from the received score from non-popular nodes

β: threshold that defines the minimum of indegree to calculate the trust

γ: the percentage of nodes that are considered as popular

N: number of iterations to calculate the iterative step of the algorithm

The Algorithm:

Step I: Initialization

1) Initializing the trust score to 0 to all the nodes

2) For the nodes having 'in degree' (The number of *edges* coming into a *vertex* in a *directed graph*) greater than 'β', calculating the mean of all the received values. (We suppose that under β in degree the mean received trustworthiness values are biased)

For nodes in U_1:

$$node_{Trustworthiness} = \frac{\sum_U received_{trust}}{Number\ of\ received\ trust\ scores};$$

Where U_1 is the set of nodes that have in degree equal, or greater, than β.

Comments:

The choice of β should depend on many features including the data itself, because the more we increase the value of β the more we get more 0 values as output of the first step.

β	**bitcoin-OTC** **Minimum % of 0 we get as output of step 1**	**BTC-Alpha** **Minimum % of 0 we get as output of step 1**
2	59.37	57.02
3	68.98	66.98
4	74.68	72.82
5	78.67	76.97
6	81.89	80.25

Figure 9: Percentage of the traders we will not calculate their mean received scores in function of β and for both platforms.

We do not consider β=1 because it is equivalent to not considering β at all.

For me, β should be, at least, 2 or 3 so we will not find biased trustworthiness. We can by mistake (Or not) receive a negative first score which may prevent others from trading with us which may exclude us from the platform even though we are trustworthy.

We insist that the hyperparameter β is highly important because it solves the problem we have discussed earlier in Section (III) of this paper. People creating new profiles and sending scores to themselves will be more affected by a higher threshold β. They will find themselves obliged to create a higher number of profiles.

Step II: Iterative part (Number of iterations = ‘N’**)**

Recalculating iteratively the scores by considering only received scores from people who have a positive trust score (The trust score is modified iteratively for each iteration, we stop after N iterations)

```
i = 1;
While i <= N:
    $U_2$ = set of nodes having strictly positive scores;
    For node in $U_1$:
        $node_{Trustworthiness} = \frac{\sum_{U_2} received_{trust}}{Number\ of\ received\ trust\ scores\ from\ U_2}$;
    i = i + 1;
```

Comments:

As we have tested in both networks: bitcoin-OTC and BTC-Alpha we have convergence after less than 5 iterations you can check Section (VI. 1)

The algorithm can run until convergence, for sure it will converge in both platforms as in section (VI. 1). But In case we want to run it on other sort of data we cannot be sure about convergence. That is why we preferred to use the hyperparameter N.

This step is highly important, and it solves some of the problems we have discussed in Section (III). In fact, when we let positive people vote we are eliminating the votes of the untrustworthy traders which is very important because untrustworthy people will tend to give false trustworthiness scores to fellow traders. We assume the iterative idea we proposed is important to rigorously classify trustworthy, untrustworthy and neutral people.

Step III: Final step

1) Ranking the nodes by their popularity in the network (Using HITS or PageRank algorithms to rank nodes). In fact, each algorithm gives a popularity score for each node. So, we can rank them according to their popularity scores.

2) Making a list L of the top ranked nodes (top γ % popularity scores of the nodes)

3) Calculating the final trust score by following these rules combined:
   - Only calculate the trustworthiness score for traders who have β, or more than β, as in degree. (As in step 1)
   - Only trust scores received from people who have positive trustworthiness score are considered in the calculation of the trustworthiness score. (As in step 2)
   - Traders who are not in the list of the top ranked nodes (That we have already prepared), only α % of their send score is retained. Otherwise traders who are in the list of top ranked nodes, 100% of their send score is retained.

To calculate the trustworthiness scores:

For node in U_1:

$$node_{Trustworthiness} = \frac{(\sum_{L\sqcap U_2} received_{trust} + \sum_{U_2 \backslash L} \alpha \times \mathrm{received}_{\mathrm{trust}})}{[Number\ of\ received\ trust\ scores\ from\ L\sqcap U_2 + \alpha \times (Number\ of\ received\ trust\ scores\ from\ U_2/L)]}$$

Where U_1 is the set of nodes that have in degree equal, or greater, than β. And U_2set of nodes that have positive trustworthiness scores calculated in the step II.

Comments:

This last step is carried out to consider the social connections of the user. In fact, receiving a trustworthy score from a popular trader in the platform should have a different weight than receiving a trustworthy score from a new trader. So, this step uses the output of the two first steps, considering the same spirit of the two first steps of only considering the scores received from positive people and considering the same threshold β and adding the γ considerations.

The choice of α is not obvious, we intuitively take α = 0.70, to give more considerations for the popular nodes, but we have no intention to eliminate the votes of non-popular nodes.

As the choice of α, the choice of γ is not obvious. However, considering the Figure 2 in the comments of step 1, we know that approximately only 30% of the nodes have more than 3 received edges, that is why we considered γ=0.30.

PageRank (PR), an algorithm developed by Google Search to rank web pages in their search engine results. In fact, it is a way of measuring the importance of website pages. However, we can use it to measure the importance of nodes in other networks. [7,9]

Hyperlink-Induced Topic Search (HITS; also known as hubs and authorities) is another algorithm that rates Web pages, developed by Jon Kleinberg. So, we get two rankings of nodes (hubs ranking and authorities ranking) [8,9]

6. Analysis and application of the Algorithm

a. Convergence of the Algorithm

In fact, step 2 is iterative and depends of the number of iterations N. In this exact case, we have proven that it needs less than 5 iterations to converge for both networks (bitcoin-OTC and BTC-Alpha)

Convergence is when error (called Ɛ) becomes null. Which means nothing will changes even if we keep iterating.

Mathematical definition of the error where U is the set of nodes:

$$\varepsilon = \quad \sum_{U} |previous_{trustworthiness} - new_{trustworthiness}|$$

Number of iterations N	bictoin-OTC Ɛ	BTC-Alpha Ɛ
1	540.59	1286.72
2	69.52	157.94
3	3.41	12.27
4	0	0.62
5	0	0

Figure 10: Variation of the error of trustworthiness in function of the number of iterations N.

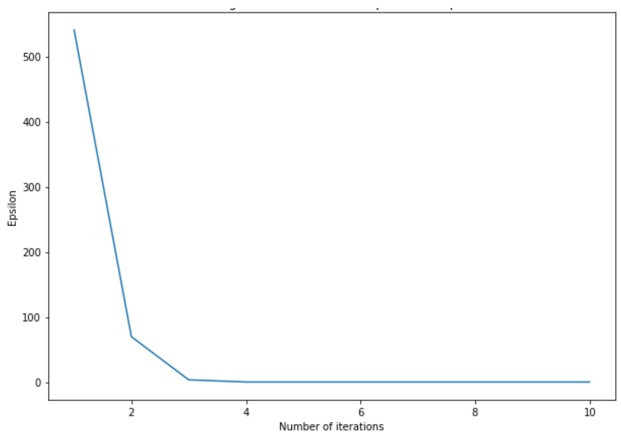


Figure 11: Convergence of the second step for BTC-Alpha platform

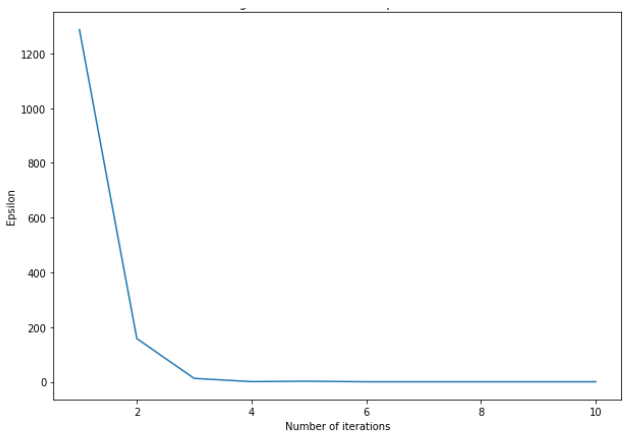


Figure 12: Convergence of the second step for Bitcoin-OTC platform

Both graphs show the variation of Ɛ as a function of the number of iterations N.

b. Robustness of the Algorithm

'Informally, robustness can be defined as the ability of a software to keep an "acceptable" behavior, expressed in terms of robustness requirements, despite exceptional or unforeseen execution conditions' [12].

We intuitively thought that if we take less edges (also we can take less nodes) we will probably get quite different results of trustworthiness values for some nodes. Indeed, when we add an edge or a node, we should get coherent results.

Supposing the ideal trustworthiness results are when we calculate considering all the data, all the nodes and all the edges. For me to consider my algorithm robust, it should have, even though we eliminate a percentage of edges, a nearby trustworthiness results. As well for me, we should find that the more we add edges the more we should find nearby trustworthiness results, so the more we get smaller mean error.

We define the mean error as ζ which shows the difference between what we get if we use the whole data to calculate the trustworthiness scores of nodes and the trustworthiness scores we get if we eliminate some edges.

$$\zeta = \sum_{U} |previous_{trustworthiness} - new_{trustworthiness}|$$

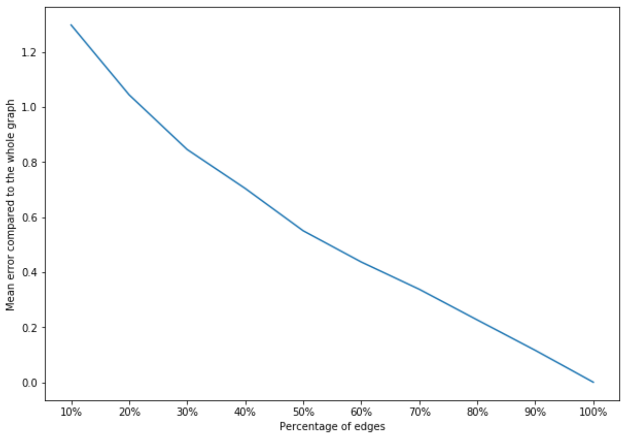


Figure 13: Variation of error for BTC-Alpha platform

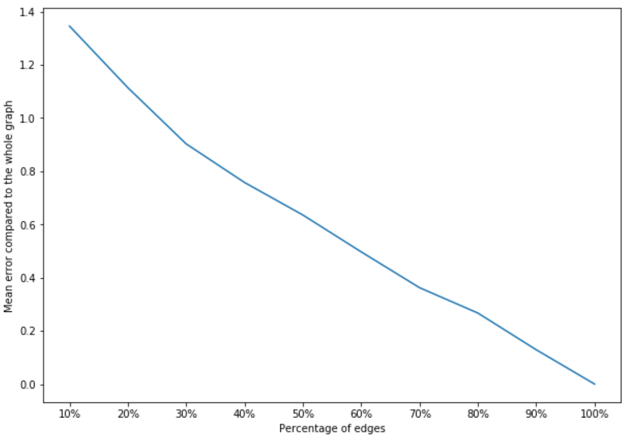


Figure 14: Variation of error for Bitcoin-OTC platform

Both graphs show the variation of ζ in function of percentage of edges we used to calculate the trustworthiness scores.

c. Correlation Analysis with features in the Network

We have calculated, for each node, many features like the in-degree, out-degree, PageRank score, hub score, auth score, mean of received trustworthiness values, mean sent trustworthiness values. Then we have seen their correlations with the output of the trustworthiness algorithm.

Features	Trustworthiness
Trustworthiness	1
Mean of all received trust scores	0.53
Hub score	0.18
Out degree	0.15
Mean of sent trust scores	0.15
In degree	0.14
PageRank	0.13
Auth score	0.12

Figure 15: The correlation between some features and Trustworthiness score estimated by the algorithm.

The mean of received trust scores is highly correlated with the trustworthiness values which is logical, because the more trustworthy we are, the more we get higher trustworthy scores form traders and the higher will be the mean.

In addition, the more we are popular on the platform, the more we have higher trustworthiness score. Because being popular on the platform means having many connections, i.e trading with many traders and being probably trustworthy.

d. Measuring Edge Scores Using Trustworthiness Estimations

We used the trustworthiness scores of nodes that we have calculated using the trustworthiness algorithm to predict the signs of the edges. The more trustworthy a person is, the more he will get better scores from other traders. We have got interesting resutls for both networks. In fact, we used logistic regression model [13]. We only used two features to make the predictions that are the trustworthiness of trustee and the trustworthiness of the trustor.

For example, for bitcoin-OTC platform, we have got an accuracy of 92%.

	precision	**recall**	**F1-score**	**support**
-	0.72	0.30	0.42	1797
+	0.93	0.99	0.96	15999

Figure 16: Analysis of the logistic regression model using two features (Trustworthiness of trustee and Trustworthiness of trustor) to predict the trustworthiness score given by the trustor to the trustee (predict score of the edge).

7. Conclusion

This algorithm can optimize the user experience on the trading platforms by labeling each one with a score that can summarize wisely his/her trustworthiness. In fact, it will help traders recognize the untrustworthy traders and the trustworthy traders. Also, it gives people the opportunity to trade even though they received some negative trust scores from spammers. In addition, this algorithm is flexible, and it depends on the proposed hyperparameters α, β, γ, and N. The choice can depend on the data, the context but can also depend on the vision of the user of this algorithm. Finally, we insist that this algorithm can be used in different domains not only the measurement of trustworthiness.

General Conclusion

This internship offered me the chance to dive through the graph theory and expand my knowledge about different social networks analysis methods.

I have been through different graph analysis online courses. Including ‘Applied Social Networks Analysis using Python’ Offred by Michigan University which helped understand deeply social networks analysis methods and its potential applications.

At the same time, I have been through different articles and papers turning over graph theory, social network analysis, machine learning, the sociology of trust, over the counter trading and bitcoin currency.

As, I have been through implementing different solutions and analysis on different social graphs which helped me master manipulating graphs and developing my programming skills using Python.

Furthermore, I had the chance to write a paper about estimating trader’s trustworthiness and I am looking forward to publishing it.

Being in meetings twice a week, working in open space, communicating with engineers and researchers and discussing applications of my project and other close projects helped me develop my understanding of different topics and most importantly develop my professional skills.

Finally, I enjoyed networking with engineers, reading papers discovering new notions, analyzing social graphs, solving real problems, bringing innovative solutions and finally summarizing all this in a scientific paper.

References

[1] S. Kumar, F. Spezzano, V.S. Subrahmanian, C. Faloutsos. *Edge Weight Prediction in Weighted Signed Networks.* IEEE International Conference on Data Mining (ICDM), 2016.
[2] *Trust as a Social Reality* J. DAVID LEWIS, *Portland, Oregon* ANDREW WEIGERT, *University of Notre Dame*
[3] *Evaluating faces on trustworthiness after minimal time exposure*, Alexander Todorov, Manish Pakrashi, and Nikolaas N. Oosterhof *Princeton University*
[4] https://wiki.bitcoin-otc.com/wiki/OTC_Rating_System

[5] https://snap.stanford.edu/data/soc-sign-bitcoin-otc.html

[6] S. Kumar, B. Hooi, D. Makhija, M. Kumar, V.S. Subrahmanian, C. Faloutsos. *REV2: Fraudulent User Prediction in Rating Platforms.* 11th ACM International Conference on Web Searchand Data Mining (WSDM), 2018.

[7] *The Google PageRank Algorithm and How It Works*, Ian Rogers IPR Computing

[8] https://en.wikipedia.org/wiki/HITS_algorithm

[9] *Applied Social Networks Analysis*, Michigan University Online Course, Coursera

[10] https://www.bitcoin-otc.com/

[11] https://btc-alpha.com/en/exchange/BTC_USD

[12] A model-based approach for robustness testing Jean-Claude Fernandez, Laurent Mounier, and Cyril Pachon

[13] https://en.wikipedia.org/wiki/Logistic_regression

[14] Counterparty Risk, Investopedia , REVIEWED BY JAMES CHEN AND CHRIS B MURPHY, RISK MANAGEMENT

[15] https://wiki.bitcoin-otc.com/wiki/Using_bitcoin-otc#Risk_of_fraud

[16] IEEE Journal on Selected Areas in Communications Volume 31 issue 9 2013 Shafiq, M. Z.; Ilyas, M. U.; Liu, A. X.; Radha, H. -- Identifying Leaders and Followers

[17] The Academy of Management Review Volume 4 issue 4 1979 [doi 10.2307%2F257851] Noel M. Tichy, Michael L. Tushman and Charles Fombrun -- Social Network Analysis for Organizations

[18] Visualising My Facebook Network Clusters – Towards Data Science

[19] Homophily - The New York Times

[20] Homophily in online dating 2005

[21] https://en.wikipedia.org/wiki/Social_network_analysis