



Ca' Foscari
University
of Venice

Single cycle degree programme

in Accounting and Finance (curriculum Business Administration)

Final Thesis

Network effects as value drivers for online digital companies

Supervisor

Federico Beltrame

Graduand

Nikolai Ostrikov

Matriculation Number 869930

Academic Year

2020 / 2021

Table of contents

List of Figures	2
List of Tables	4
Introduction	5
Chapter 1. Value drivers of a digital network company	7
1.1 Networks – key characteristics and network effects	7
1.2 Network dynamics: growth, evolution & resilience	13
1.3 SIR and SIS models of network epidemic spreading	14
1.4 SIS and SIR epidemics on scale-free and real-world networks	18
1.5 Social perspective on network effects	19
1.6 Business models of online digital network companies	22
Chapter 2. Valuation of digital network companies	30
2.1 Methodology	30
2.2 Valuations	35
2.2.1 XING	35
2.2.2 Facebook	41
2.2.3 Adapting SIR and SIS models for more accurate forecasting using data	44
Chapter 3. Empirical analysis of the quality of annual reports	47
3.1 Similarity analysis of annual reports	47
3.2 Lexical analysis of annual reports	49
Conclusion	53
Appendix A. Python code – Chapter 1	54
Appendix B. Python code – Chapter 2	66
Appendix C. VBA code for generating triangular distributions in MS Excel	68
Appendix D. Cosine similarity matrixes by companies	69
Appendix E. Cosine similarity matrixes across companies by years	73
Appendix F. Python code – Chapter 3	75
References	81

List of Figures

Figure 1. Degree histogram Path graph	9
Figure 2. Degree histogram Star graph	9
Figure 3. Degree histogram Complete graph	9
Figure 4. Degree histogram Facebook sample Graph	11
Figure 5. Degree histogram Twitter sample graph	11
Figure 6. Degree histogram Lifejournal sample graph	11
Figure 7. Snapshot of SIR on path graph at $t=100$ (transition rate = 0.5, recovery rate = 0)	16
Figure 8. Distribution of the quantity of infected nodes at $t=100$ (10 000 simulations)	16
Figure 9. 500 SIR simulations on a path graph (transition rate = 0.7, recovery rate = 0)	16
Figure 10. SIR simulation – Random network	17
Figure 11. SIS simulation – Random network	17
Figure 12. Barabasi-Albert graph ($m=1$)	17
Figure 13. Barabasi-Albert graph ($m=3$)	17
Figure 14. Barabasi-Albert graph ($m=5$)	17
Figure 15. SIR on Barabasi-Albert graph ($m = 1$)	17
Figure 16. SIR on Barabasi-Albert graph ($m = 3$)	17
Figure 17. SIR on Barabasi-Albert graph ($m = 5$)	17
Figure 18. SIS on Barabasi-Albert graph ($m = 1$)	17
Figure 19. SIS on Barabasi-Albert graph ($m = 3$)	17
Figure 20. SIS on Barabasi-Albert graph ($m = 5$)	17
Figure 21. 100 SIR simulations on 10 scale-free Barabasi-Albert graphs	18
Figure 22. 100 SIR simulations on 10 scale-free Barabasi-Albert graphs (normalized)	18
Figure 23. 100 SIS simulations on 10 scale-free Barabasi-Albert graphs	18
Figure 24. 100 SIS simulations on 10 scale-free Barabasi-Albert graphs (normalized)	18
Figure 25. SIR on Twitter	19
Figure 26. SIR on Twitter (normalized)	19
Figure 27. SIR on LiveJournal	19
Figure 28. SIR on LiveJournal	19
Figure 29. SIS on Twitter	19
Figure 30. SIS on Twitter (normalized)	19
Figure 31. SIS on LiveJournal	19
Figure 32. SIS on LiveJournal (normalized)	19

Figure 33. Business model canvas - Freemium business model	23
Figure 34. Business model canvas – Advertising business model of Facebook	24
Figure 35. Business model canvas – YouTube and Google business model	27
Figure 36. Business model canvas – LinkedIn business model	28
Figure 37. Modelling user-retention rate using the calibration factor α in the model of Gneiser et al. (2012)	32
Figure 38. SIR process in Facebook	34
Figure 39. SIR process in Twitter	34
Figure 40. SIR process in LifeJournal	34
Figure 41. SIS process in Facebook	34
Figure 42. SIS process in Twitter	34
Figure 43. SIS process in LifeJournal	34
Figure 44. XING intrinsic value (EUR/share) distribution	37
Figure 45. Forecasting quantity of XING users using SIR simulations	39
Figure 46. Facebook intrinsic value distribution (base)	42
Figure 47. Facebook intrinsic value distribution (SIR Simulation-based)	43
Figure 48. Forecasting quantity of Facebook users with SIR simulations	44
Figure 49. Lexical dispersion plot - Amazon	50
Figure 50. Lexical dispersion plot - Activision Blizzard	50
Figure 51. Lexical dispersion plot - Ebay	50
Figure 52. Lexical dispersion plot - Facebook	50
Figure 53. Lexical dispersion plot - Google	50
Figure 54. Lexical dispersion plot - LinkedIn	50
Figure 55. Lexical dispersion plot - Mail.ru group	50
Figure 56. Lexical dispersion plot - Microsoft	50
Figure 57. Lexical dispersion plot - Match group	51
Figure 58. Lexical dispersion plot - Royal Dutch Shell	51
Figure 59. Lexical dispersion plot - Snapchat	51
Figure 60. Lexical dispersion plot - Twitter	51
Figure 61. Lexical dispersion plot - XING	51
Figure 62. Lexical dispersion plot - ExxonMobil	51
Figure 63. Lexical dispersion plot – Yandex	51

List of Tables

Table 1. Descriptive statistics of the 3 basic 10-node graphs	10
Table 2. Spreading parameters for sample SIR and SIS simulations	16
Table 3. Inputs for DCF	35
Table 4. XING revenues by segment (2015-2019)	36
Table 5. XING EBITDA margins by segment (2015-2019)	36
Table 6. XING B2C segment revenue simulation parameters	36
Table 7. XING B2B segment revenue simulation parameters	37
Table 8. XING marketing segment revenue simulation parameters	37
Table 9. Descriptive statistics of XING's intrinsic value distribution (base case)	38
Table 10. Descriptive statistics of XING's intrinsic value distribution (B2C forecast using SIS)	39
Table 11. Facebook key financials 2013-2019	41
Table 12. Facebook revenues simulation parameters	41
Table 13. Descriptive statistics of Facebook's intrinsic value distribution (base case)	42
Table 14. Descriptive statistics of XING's intrinsic value distribution (B2C forecast using SIS)	44
Table 15. Company reports selected for textual analysis	47
Table 16. Average report similarity coefficients by years	48
Table 17. Regression outputs - Changes in stocks' betas and the cosine similarity of companies' reports	49

Introduction

Valuation of young growth companies is a challenging task – they not only don't have a long story, but also often operate in new businesses. Moreover, business models of the new growth companies often exploit various network effects which makes the behavior of their financials non-linear and confusing to the market participants. As a consequence, the stock price of such type of companies is highly volatile and the probability of being valued wrong by the market increases since the market relies on a wrong set of signals – in the dotcom bubble it relied mostly on website visitors, and now it tends to rely on the quantity of users. Current company information disclosure standards do not allow to construct reliable user-based company valuation models. As a consequence, top-down aggregate approach to valuation has been commonly used mainly due to its simplicity given that disaggregate approaches to valuation and pricing gave comparable or higher uncertainty about the company value estimates while bringing in the model a huge number of assumptions about unit economics of the businesses.

This research is an attempt to extend the intrinsic value and pricing approaches to value a user in a network. It will focus on a set of young growth companies which significantly rely on network effects in their business models and which have to expose information about networks of their users (or at least part of it) by design of their business models. Recent developments in the network theory and availability of cheap computing power and publicly available data may allow to increase the accuracy of DCF valuation models for this type of companies. For companies like Facebook, LinkedIn, Xing and other online social networks data about their user-base (or at least some part of it) is publicly available. For example, the head of data-analytics company Tazer Global claimed that nowadays any student with a laptop can scrape the data of the whole online social network in one week (basic user profile data, friendships and posts) and the whole dataset of the Russian segment of Facebook would occupy less than 100 Gb of space (Hachuyan 2017). However, in order to assess the value of a network it might not be necessary to have a dataset which represents the whole network. Sometimes companies disclose datasets themselves when they run Hackathons and challenges from which they hope to crowdsource new algorithms to extract more value from their user data. In addition, the data obtained from various data breaches could also be used as a source for building such models regardless of the ethical and legal issues. Facebook, LinkedIn and other existing online social networks have already mapped significant portion of the real-world social relationships and given that the Facebook has a user base which reaches 90% in some of the regions and countries it operates in it has become possible to use the user data to predict how other network companies business models would perform in such networks. Therefore, in order to build disaggregated user-based valuations of companies which

don't expose their networks to public like Uber or Lyft one could use Facebook's or LinkedIn's graphs enhanced with other information one can attach to it, or, alternatively, a simulated graph which reproduces the main qualities of the real-world network if interest.

It will be examined whether the complexity of bringing the network theory instruments in the valuation and pricing models adds value to them by exploring the tradeoff between additional complexity and the plausibility of assumptions vs. increased precision of the valuation.

This research consists of 3 chapters: chapter 1 is devoted to the basic qualities of networks and processes occurring on them, chapter 2 deals with methodological and practical aspects of valuation of online network companies. In the final chapter the companies' disclosure standards are being analyzed using computer linguistic and statistical techniques. Findings and simulation algorithms developed as the result of this research will be of a value to researchers and practitioners of business valuation and allow to generate more accurate inputs for valuation models.

Chapter 1. Value drivers of a digital network company

1.1 Networks – key characteristics and network effects

For a proper bottom-up valuation of network companies it is necessary to briefly review the basic concepts and statistical tools of network theory. The network science is a relatively young discipline. This research relies on the systematizations in the field by Barabási and Newman. The following notation is used: a network is a set of nodes. Nodes of a network share connections that may take form of edges or hyperedges (Newman 2003). Edges represent connections between two nodes and may be either directed from one node to another or undirected, while hyperedges connect more than two nodes and are undirected (Newman 2003). Nodes may carry additional quantitative and qualitative parameters. Edges may be assigned with weights which allow to quantify the relationships among edges in the network model. The same nodes may simultaneously be included in several different networks. For example, in a real-life social network each person (node) has separate sets of professional and private contacts (edges) and, at the same time, is associated with various social groups and organizations (hyperedges). The notion of a node also depends on the design of the research model. Nodes may as well represent groups of people that share a common characteristic of interest (nationality, age, gender, location, income etc.), therefore, broadly, a node may be defined as an actor in a network and the edges and hyperedges – as relationships among those actors. The same real-life network may be modeled in several ways: 1. using only nodes and edges or 2. using nodes, edges and hyperedges. In the first case the nodes would represent a complex object: for example, a group of people living in the same city, country etc. If one deems necessary for the nodes to represent minimal unit of the network then a different model would be constructed: nodes would represent individuals instead of groups, edges would bear the same function as in the previous case, and hyperedges would embed in the model the grouping by city, country, etc.. The use of hyperedges brings multidimensionality in the model and makes the analysis more complex. However, multidimensionality of a network allows to account for all the necessary factors simultaneously while in the network models that use only nodes and edges are focused on a particular parameter which is used to group the atoms of a network to make up a node. Another way to avoid using hyperedges is to use nodes of different types (movies and actors, chemical elements and nutrition products etc.). The networks which include several types of nodes are called multipartite (Barabási 2016). In some networks nodes may have more than one edge between each other and this type of graph is called multigraph. In some network models nodes may be allowed to interact with themselves which is represented by self-loops – edges, which are connected only to one node. The choice of the network model has to be aligned with the research objectives. This research is focused on online social network

companies, therefore (unless specified otherwise) the nodes represent people and edges/hyperedges represent relationships among the people.

The qualities of simple networks with a low number of nodes and edges may be analyzed using graphical visualizations, that allow to explore all possible paths in networks and get exact values for various parameters that characterize a network (Newman 2003). However, real-world networks are much more complex. They are made up by billions of nodes and edges/hyperedges consequently analysis of their structure and processes which occur in them is only possible through the use of the graph theory and statistical techniques (Newman 2003).

Nodes, edges, hyperedges of networks possess quantitative characteristics. The number of connections of a node (k_i) is a degree of a node. Total number of connections in a network (L) is $\frac{1}{2}$ of the sum of degrees of all the nodes in the network (Eq. 1.1).

$$L = \frac{1}{2} \sum_{i=1}^N k_i \quad (\text{Barabási 2016}) \quad (1.1)$$

For any network the total number of connections lies between 0 and L_{max} determined by Eq. 1.2.

$$L_{max} = \frac{N(N-1)}{2} \quad (\text{Barabási 2016}) \quad (1.2)$$

If the network has maximum number of connections it may possibly have – every node is connected to every other node in the network, it is called a complete graph.

One of the measures of interconnectedness of a network is the average degree \bar{k} of a network:

$$\bar{k} = \frac{1}{N} \sum_{i=1}^N k_i = \frac{2L}{N} \quad (\text{Barabási 2016}). \quad (1.3)$$

The average degree in a network with N nodes may vary from 0 for the networks with no edges which are called empty (or null) graphs to $N - 1$ for complete graphs in which every node has connections with all other nodes.

Another measure of interconnectedness of a network is the network density, which is the ratio of the edges present in the network to the maximum potential number of edges the network can have given its number of nodes. Network density can be calculated using the formula:

$$D = \frac{L}{L_{max}} \quad (\text{Barabási 2016}) \quad (1.4)$$

Edges in a graph may represent one-way relationships and make up a directed graph. In directed graphs degree of a node equals to the sum of its incoming degree (nodes that point to the node) and outgoing degree (number of nodes that the node is pointing to).

Degree distribution is one of the key properties of networks. In graph theory it is described by a normalized probability function p_k which expresses the dependency of the probability of a randomly chosen node in a network to have a degree equal to k .

$$\sum_{k=1}^{\infty} p_k = 1 \quad (\text{Barabási 2016}) \quad (1.5)$$

$$p_k = \frac{N_k}{N} \quad (\text{Barabási 2016}) \quad (1.6)$$

The three possible extremes in degree distributions are illustrated on the Figures Figure 1Figure 3 using networkx package for Python (Hagberg, Swart and Chult 2008).

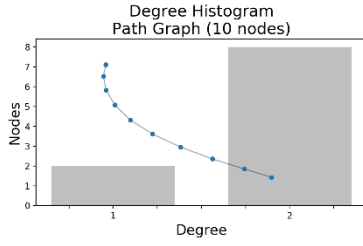


Figure 1. Degree histogram Path graph

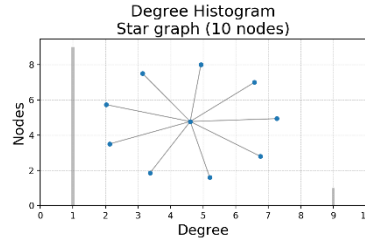


Figure 2. Degree histogram Star graph

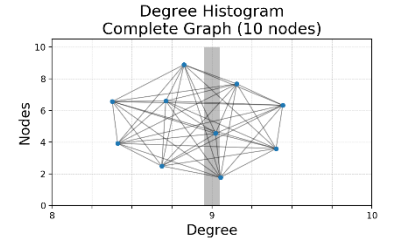


Figure 3. Degree histogram Complete graph

The first possible extreme is a path graph – all nodes have a degree of two and two nodes at the ends of the graph have a degree of one. The second one is a “star” graph – all nodes are connected to one hub, therefore all nodes have degree equal to one and the hub’s degree is equal to nine. The third possible extreme is a complete graph – all nodes are connected to all other nodes and have a degree of 9 (N-1).

Network path is a chain of edges from one node to another. The path length is equal to number of edges it contains. The shortest path between two nodes is the one which has the least number of edges. In a directed network the existence of a path from a node x to a node y does not guarantee the existence of a path from node the y to the node x. Network diameter is the longest path that exists in a network. In small networks the network diameter is easy to calculate, while in large networks with millions of nodes this becomes an non-trivial task for which various algorithms have been developed: breadth-first search (Barabási 2016), navigation algorithm (Liu et al. 2019), Dijkstra’s algorithm (Dijkstra 1959) and others. The average of all shortest paths between all pairs of nodes is called average path length of a network.

Networks may be composed of several sub-networks which are not connected to each other. These sub-networks are called connected components. For each connected component clustering coefficients of nodes reflect how even connections are spread. The local clustering coefficient may be calculated using the formula:

$$C_i = \frac{2L_i}{k_i(k_i-1)} \text{ (Barabási 2016)} \quad (1.7)$$

where L_i is the sum of degrees of the nodes which are connected to the node i. The value of the clustering coefficient lies between 0 (the neighbors of the node i don’t share connections with each other) and 1(the neighbors of the node i are all interconnected and form a complete graph). The local clustering coefficient may also be viewed as a local network density measure (Barabási 2016). The clustering of the whole network can be measured by calculating average clustering coefficient by the formula:

$$\bar{C} = \frac{1}{N} \sum_{i=1}^N C_i \text{ (Barabási 2016)} \quad (1.8)$$

Uneven clustering results in emergence of locally dense structures in a network which are called communities. It is possible to define communities in two ways – strong and weak. Strong communities are those in which each of the nodes has more connections with other community members than with the rest of the network. Weak communities are subgraphs in which internal to external cumulative degree ratio is greater than 1 (Barabási 2016). Community detection is a very complicated task since in order to be able to detect the best partitioning one has to analyze all possible combinations of communities in a graph which grows exponentially to the size of a network. Instead, there are two types of algorithms which can be used to complete the community detection task in feasible time with a certain probability of the best slicing of the network: agglomerative and divisive algorithms. Agglomerative algorithms find similarities in groups of nodes and merge them into communities (for example, Ravasz algorithm (Ravasz et al. 2002) or link clustering algorithm (Ahn, Bagrow and Lehmann 2010, Evans and Lambiotte 2009)). Divisive algorithms, like Girvan-Newman Algorithm (Girvan and Newman 2002, 2004), are meant to detect the least similar connections and remove eliminate them to obtain communities. The criteria of similarity vary from one algorithm to another. The community partition problem can also be approached by clustering links instead of nodes based on their topological similarity (Barabási 2016). The main application of the community identification algorithms is identification of customer groups and their interests in order to align marketing and sales strategies of companies. Network density, average degree, average shortest path length, network diameter and average clustering coefficient are the basic descriptive statistics for a graph. In Table 1 these statistics are computed for the 3 sample 10-node graphs pictured on Figures Figure 1Figure 3. In the Table 2 the same statistics have been computed for the research datasets for three popular online social networks – (Facebook, Twitter, LifeJournal). Figures Figure 4-Figure 6 contain their graphical representation and degree distribution histograms.

Table 1. Descriptive statistics of the 3 basic 10-node graphs

	Path graph (Figure 1)	Star graph (Figure 2)	Complete graph (Figure 3)
Number of edges	8	9	90
Network density	0.088	0.100	1.000
Average degree	1.6	1.8	18
Average shortest path length	3.67	1.8	1
Network diameter	9	2	1
Average clustering coefficient	0	0	1

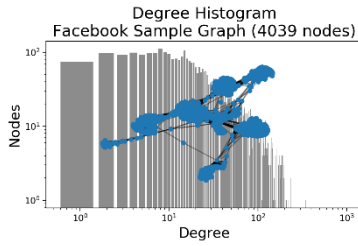


Figure 4. Degree histogram Facebook sample Graph

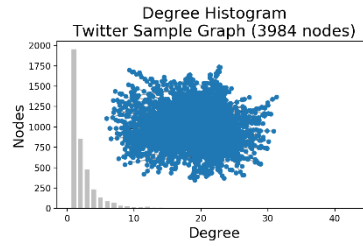


Figure 5. Degree histogram Twitter sample graph (McAuley and Leskovec, 2012)

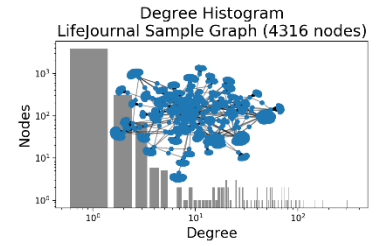


Figure 6. Degree histogram Lifejournal sample graph

Table 2 - Descriptive statistics of social network sample datasets

	Facebook	Twitter	LifeJournal
Number of nodes	4 039	81 306	4 847 571
Number of edges	88 234	1 768 149	68 993 773
Network density	0.0108	0.0004	0.0703
Average degree	43.69	33.01	22,7
Average shortest path length	3.6925	9.92	3.88
Network diameter	8 (4,7*)	7(4,5*)	16(6.5*)
Average clustering coefficient	0.6055	0.5653	0.2742

*90-th percentile effective diameter (McAuley and Leskovec, 2012)

From the Table 2 it can be seen that even in huge real-world social network samples the network diameter is incomparable to network size and 90 percent of the nodes are separated only by a 4-7 edges. This is in line with “Small world effect”, which has been observed in numerous studies (Milgram 1967, Travers and Milgram 1969, Pool and Kochen 1978, Albert, Jeong and Barabási 1999, Lawrence and Giles 1999). The existence of the small world effect in every particular network depends on the network formation mechanisms that define the attachment of nodes.

The first type of network formation mechanisms is random attachment, which produces random networks. In the literature there are two main definitions of a random graph: “G(N, L) Model: N labeled nodes are connected with L randomly placed links. (Erdős and Rényi 1959) and “G(N, p) Model: Each pair of N labeled nodes is connected with probability p” (Gilbert 1959). The degree distribution in a random network is binomial, but for networks in which the average degree is small compared to the quantity of nodes the binomial distribution can be approximated by Poisson distribution. The benefit of approximation – while the binomial distribution is characterized by two vectors – vector of probabilities p_k and vector of quantity of nodes for each p_k which correspond to each probability N, the Poisson distribution depends is fully characterized by

average degree $\langle k \rangle$ and number of nodes N (Barabási 2016). In a random network with average degree $\langle k \rangle$ and number of nodes N the diameter of the network can be approximated as:

$$\langle d \rangle \approx \frac{\ln N}{\ln \langle k \rangle} \quad (\text{Barabási 2016}) \quad (1.9)$$

This formula illustrates that the higher the average degree the shorter would be the average path length in network. And, on the other hand, the average shortest path length grows proportionally to logarithm of its number of nodes or, disproportionally slower. The disproportional relationship between the network diameter and the number of nodes leads to existence of what Milgram coined as famous “6 degrees of separation”(1969) or Pool and Kochen’s concept of “small world” (1978). Experiments confirm the validity of these calculations (Travers and Milgram 1969, Backstrom et al. 2012, Edunov et al. 2016). The degree distribution of a random network does not depend on the quantity of nodes, which makes random networks “scalable” (Barabási 2016). However, the observed degree distributions and of real-world networks do not correspond to those predicted by random models, which don’t allow existence of high-degree outliers or “hubs” (Rosenthal 1960, Freeman and Thompson 1989, Gjoka et al. 2010, Ugander 2011, Catanese et al.2011, Watanabe and Suzumura 2013, Myers et al 2014). For instance, random network models do not account for skewness of degree distribution, which can be seen in most of the real-world networks. If the real-world network of social relationships among people had followed Poisson distribution the majority of people would have had social interactions with 968 to 1032 persons and there wouldn’t have been outliers who have much more connections (expected maximum degree for 7 bln. Poisson network would be 1185) (Barabási 2016). According to all of the mentioned researches the degree distributions of the the real-world networks is closer to follow power law rather than being random. Power law distribution is defined by the following formula $p_k \sim k^{-\gamma}$ where γ is the degree exponent (Barabási 2016). From this formula it is possible to conclude that $\log p_k$ is a linear function of $\log k$ with a slope of $-\gamma$. A network distribution of which follows power law is scale-free. In contrast with a random Poisson distribution in which all nodes degrees always stay within half of the mean range, which makes all random networks to have comparable degrees, “scale” of which depends solely on the mean degree $\langle k \rangle$ (Barabási 2016). In contrast, in a scale-free network the standard deviation (second moment of degree distribution) is infinite, therefore a randomly chosen node can have any scale. Power law degree distribution’s major difference from a random one is the long tail – it allows the existence of hubs – extremely high-degree nodes. For many real-life scale-free networks the observed degree exponent lies between 2 and 3.

The scale-free property leads to the friendship paradox – for each of majority of nodes in a scale-free network the node’s degree is less than the average degree of its neighbors or, as Feld put it “Why your friends have more friends than you do” (1991). The friendship paradox can be

explained statistically: a randomly chosen node in a scale-free network has a higher probability of being connected to one or more “hubs” that have higher degree rather than being connected to a large number of low-degree nodes.

In the real world the process of network formation is shaped by various natural conditions: geography (for example, internet infrastructure (Yook, Jeong, and Barabási 2002)), social factors (Doreian and Conti 2012.) and others. The growth and structuring of online social networks is primarily shaped by geographical, legislative, national, political, cultural, linguistic, economical, technological and demographical factors.

1.2 Network dynamics: growth, evolution & resilience

At this point the most important characteristics of networks have been discussed, so it is time to briefly recap the theory of dynamic processes in networks – network growth, network evolution, network resilience, cascading failure and network spreading.

Essentially the process of network formation and growth is attachment of new nodes to existing components of a network. For a random network a new node has equal probability to establish connection with any other node in the network while non-random networks form as a result various forms of preferential attachment. The preference function defines the probability for a new node to establish connection with other nodes in a network. One possible way for a scale-free network to form is to distort the probability of a new node to connect to other nodes in favor of those which have higher degree. As a result, the degree distribution will follow power law (Barabási 2016). Together with growth real-world networks evolve with time – some nodes may be removed, others may establish new connections. These processes may lead to significant changes in clustering and community structure of the network and as a consequence – different functional characteristics.

Network resilience can be described as a probability density function:

$$\frac{P_{\infty}(f)/P_{\infty}(0)}{f} \quad (\text{Barabási 2016}) \quad (1.10)$$

where the numerator is the ratio of the probability of a randomly chosen node to belong to the giant connected component after removal of the fraction of nodes equal to f $P_{\infty}(f)$ to the same probability before percolation occurs. Scale-free networks with degree exponent greater than 3 are resilient to random node removal (it is necessary to destroy the whole network to split the giant component). The resilience to random node removal of scale-free networks is not always the case for real-world networks. For networks in which nodes and edges have a limited transmission capacity the changes of shortest network path lengths leads to overload of the remaining nodes. The overload, in turn, may result in cascading removal of other nodes (for example, in power grids (Kosterev, Taylor and Mittelstadt 1999)). Scale-free networks are also vulnerable to selective removal of hubs (Barabási 2016).

1.3 SIR and SIS models of network epidemic spreading

The dynamics of spreading processes on networks can be modeled using the epidemic spreading models SI, SIR & SIS and their modifications. The simplest model of network spreading is Susceptible-Infected (SI) model. In this model nodes of a network can be in two states – receptive to the spreading process (“Susceptible”) and activated and able to activate others through edges (“Infected”). Under the assumption that infected nodes are homogeneously distributed in a random network numerically the model is defined by the following differential equations:

$$\frac{\Delta s}{\Delta t} = -\beta i s \quad \frac{\Delta i}{\Delta t} = \beta i s \quad (\text{Newman 2003}) \quad (1.11)$$

Where s and i are fractions of susceptible and infected nodes in the network correspondingly and β is a probability of a susceptible individual to become infected as a result of a contact with another infected node.

SIR model is obtained by adding the possibility of a node to be recovered/removed from the network from the perspective of the spreading process (it may happen in two forms: a node is either removed from a network or the node becomes deactivated and immune to the spreading process). The fractions of susceptible, infected and recovered nodes at any time during the spreading process may be found using the following differential equations:

$$\frac{\Delta s}{\Delta t} = -\beta i s \quad (1.12)$$

$$\frac{\Delta i}{\Delta t} = \beta i s - \gamma i \quad (1.13)$$

$$\frac{\Delta r}{\Delta t} = \gamma i \quad (\text{Newman 2003}) \quad (1.14)$$

The introduction of removal (or, equally, recovery with immunity) to the model means that the only possible outcome of such spreading process would be infection with subsequent recovery/removal of all nodes in the connected component of the random network.

In case if a recovered node can be activated again the SIS model should be applied and the equations would be the following:

$$\frac{\Delta s}{\Delta t} = -\beta i s + \gamma i \quad (1.15)$$

$$\frac{\Delta i}{\Delta t} = \beta i s - \gamma i \quad (\text{Newman 2003}) \quad (1.16)$$

As the result of the spreading process there are two possible outcomes: first would be - the system enters endemic state in which the ratio of infected nodes in the network stabilizes at a non-zero level meaning that the quantity of recovered is fully balanced by the quantity of infected for each Δt . The second outcome, when the number of infected nodes steadily goes to 0, occurs if the spreading process has β and γ such that it does not pass epidemic threshold (Newman 2003).

If the spreading process takes place on a non-random network the SI, SIR and SIS models have to be modified. It is possible to adjust the equations to take into account degree distributions (Grassberger 1983, Sander et al. 2002, Newman 2002, Moreno 2003). It is also possible to derive that for most of the scale-free networks with the degree exponent greater than 3 (which most real-life networks are) the epidemic threshold approaches zero no matter the transmission probability and recovery rate of the process (Boguá, Pastor-Satorras and Vespignani 2003) excluding some particular types of networks with specific correlations among node degrees (Boguná and Pastor-Satorras 2002, Boguá, Pastor-Satorras and Vespignani 2003, Blanchard, Chang and Krüger 2003, Moreno and Vazquez 2003), meaning that in most of the real-world networks the spreading processes can be sustainable not because of their characteristics, but instead due to characteristics of a specific network they occur in. It is not possible to derive the full analytical model for each network it is necessary to analyze. Instead, in this research Monte-Carlo simulations of the network spreading are used in order to be able to include all the necessary aspects of the spreading process and qualities of a network into model. The aspects and qualities, which may be taken into account includes, but not limited to: conditional recovery rates, transmission probabilities driven by weights (or node sizes), multiple spreading processes and their competition (or cooperation), random/selective immunization. All SIS and SIR simulations in this research are performed using EoN (epidemics on networks) package for Python (Miller et al. 2019).

In order to get a basic understanding of how the EoN package works one of the simplest functions - EoN.fast_SIR is tested on a path graph (a chain of 100 nodes). Full python code of the test is provided in the Appendix A. The experiment goes as follows: one node on the edge of a 100-long path graph is infected at the beginning of the spreading process and the simulation is run for 10 000 iterations. Figure 7 illustrates a possible state of the graph at time 100. Figure 8 illustrates the distribution of outcomes of 10 000 simulations of a SIR spreading on a path graph with spreading probability equal to 0.6 and recovery rate equal to 0. The distribution is slightly skewed to the right and does not pass strict normality tests ($p < 0.05$) (skewness is 0.11, kurtosis is -0.05).

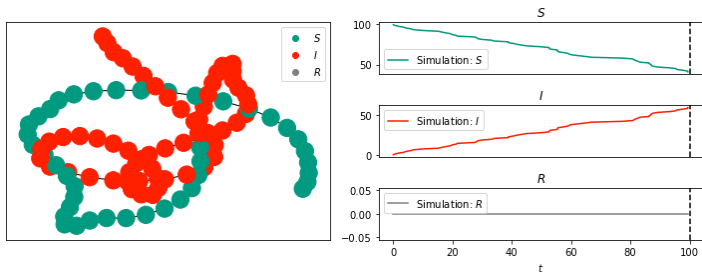


Figure 7. Snapshot of SIR on path graph at $t=100$ (transition rate = 0.5, recovery rate = 0)

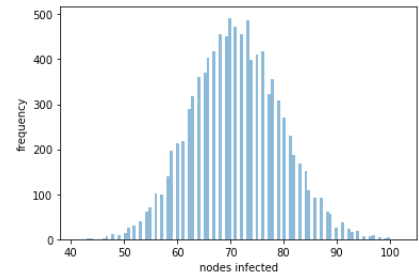


Figure 8. Distribution of the quantity of infected nodes at $t=100$ (10 000 simulations)

Figure 9 shows the results of 500 SIR simulations on a path graph with 1 node at one of the ends of the path graph initially infected. From the figure we can see that the probability of infection (in this case 0.7) means that for a path graph the number of infected nodes at time 100 is a random variable with a mean around 70 and standard deviation of 8.

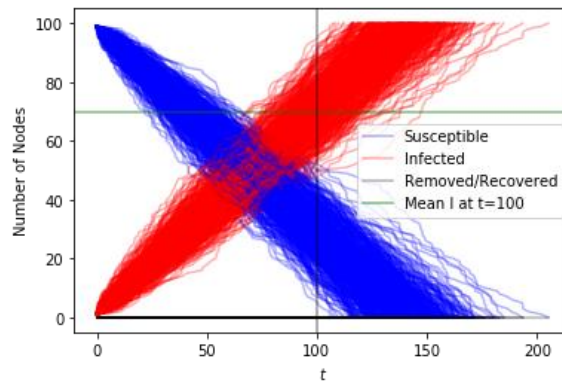


Figure 9. 500 SIR simulations on a path graph (transition rate = 0.7, recovery rate = 0)

Figure 10 and Figure 11 illustrate SIR and SIR processes on a random network of 100 nodes with average degree equal to 50 and the spreading dynamics is defined by parameters listed in Table 3.

Table 2. Spreading parameters for sample SIR and SIS simulations

Parameter	Value	Description
β	0.1	Transmission rate
γ	0.09	Recovery rate
$I(0)$	5	Nodes initially infected

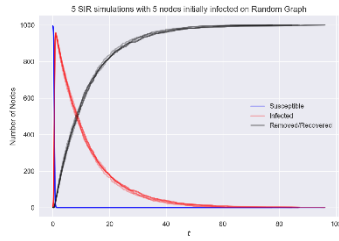


Figure 10. SIR simulation – Random network

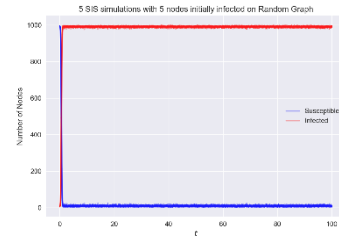


Figure 11. SIS simulation – Random network

Figures 15-20 demonstrate how the same spreading process would behave on scale-free networks of 1000 nodes (Figures 12-14 contain their degree histograms and graphical representation).

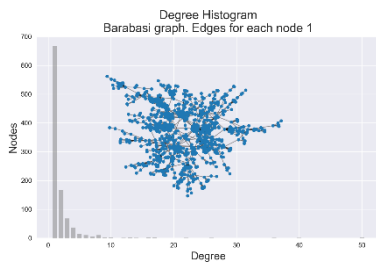


Figure 12. Barabasi-Albert graph (m=1)

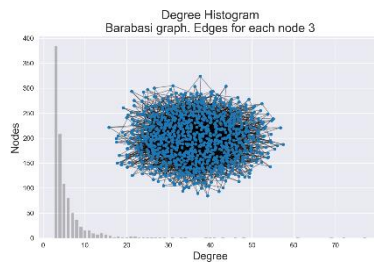


Figure 13. Barabasi-Albert graph (m=3)

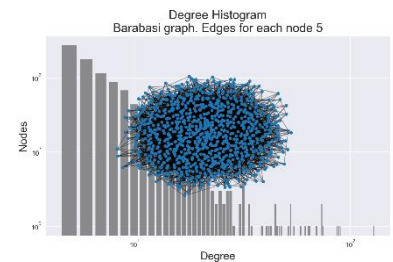


Figure 14. Barabasi-Albert graph (m=5)

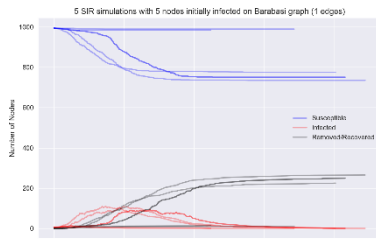


Figure 15. SIR on Barabasi-Albert graph (m = 1)

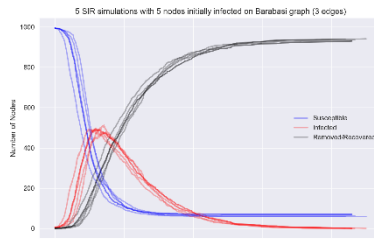


Figure 16. SIR on Barabasi-Albert graph (m = 3)

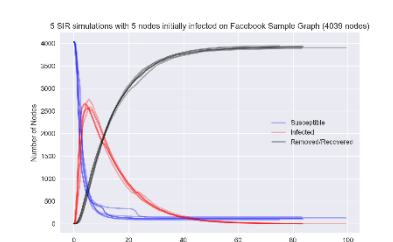


Figure 17. SIR on Barabasi-Albert graph (m = 5)

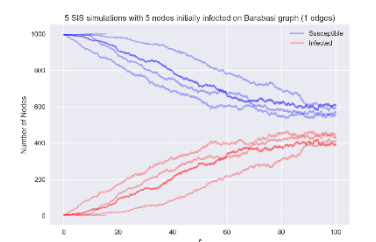


Figure 18. SIS on Barabasi-Albert graph (m = 1)



Figure 19. SIS on Barabasi-Albert graph (m = 3)

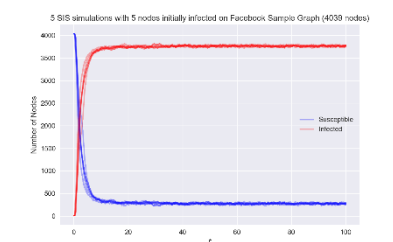


Figure 20. SIS on Barabasi-Albert graph (m = 5)

Comparing the dynamics of the spreading on Figures Figure 15-Figure 20 it is evident that the more dense a network is the more stable the spreading process is and the shorter time it takes for the system to enter the endemic state.

1.4 SIS and SIR epidemics on scale-free and real-world networks

Scale-free networks have another important property – the rate of infection as a share of the network size does not depend on the size of the network and the resulting shares of Susceptible, Infected and Removed (both in SIS and SIR models) for scale-free networks which follow the same degree distributions remain the same. Figures Figure 21 and Figure 23 demonstrate SIR and SIS spreading on Barabasi-Albert graphs of different sizes. Figures Figure 22 and Figure 24 show the same processes but the Susceptible, Infected and Removed are pictured as the share of size of networks. It is clear that the outcome of simulations tends to be the same in terms of S/N, I/N and R/N ratios regardless of the network size.

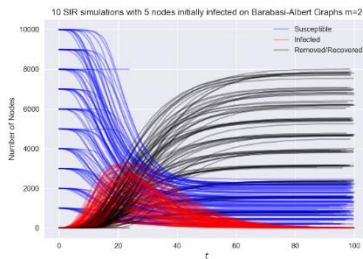


Figure 21. 100 SIR simulations on 10 scale-free Barabasi-Albert graphs

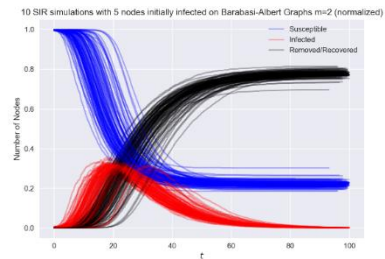


Figure 22. 100 SIR simulations on 10 scale-free Barabasi-Albert graphs (normalized)

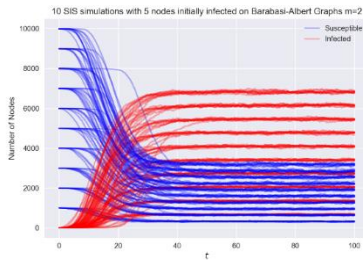


Figure 23. 100 SIS simulations on 10 scale-free Barabasi-Albert graphs

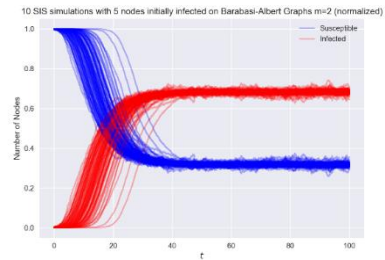


Figure 24. 100 SIS simulations on 10 scale-free Barabasi-Albert graphs (normalized)

This property of scale-free networks is remarkable - using a small sample which reproduces general network structure it is possible to relatively accurately forecast behavior of a spreading process on a scale-free network of any size. Figures 25-32 demonstrate that the same scaling effect holds for SIS and SIS processes with the same parameters on Twitter and LiveJournal graphs.

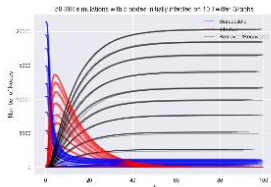


Figure 25. SIR on Twitter

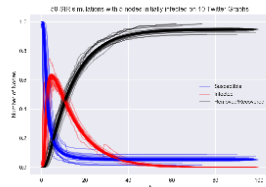


Figure 26. SIR on Twitter (normalized)

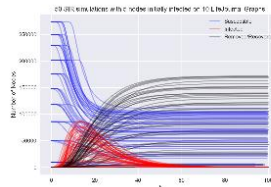


Figure 27. SIR on LiveJournal

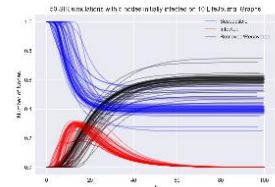


Figure 28. SIR on LiveJournal

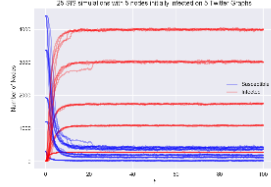


Figure 29. SIS on Twitter

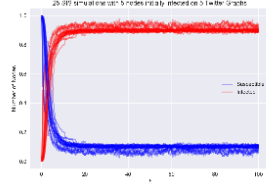


Figure 30. SIS on Twitter (normalized)

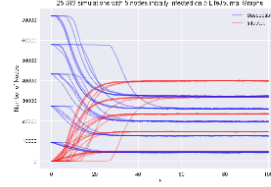


Figure 31. SIS on LiveJournal

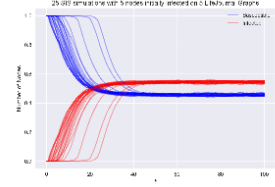


Figure 32. SIS on LiveJournal (normalized)

1.5 Social perspective on network effects

Social sciences also provide a perspective on the value sources of networks made up with social relationships among people. The concept of social capital was introduced by Lyda Hanifan in 1916. She defined it as: “Those tangible assets count for most of daily lives of people: goodwill, fellowship, sympathy, social intercourse among individuals and families who make up a social unit” (Hanifan 1916). The social bonds may as well function as distribution network of various benefits (economic, social, emotional etc.) “especially in countries where rule of law is weak” (OECD Insights: Human Capital 2007). Individuals use their access to information and influence through social networks to compete for a better position in society.

Social capital emerges from relationships among people is as a set of network effects. Early researches of Facebook have found that being active on Facebook is correlated with bonding, bridging and high school social capital (Ellison, Steinfield and Lampe 2006).

There has been an opinion that contemporary society is losing social capital. According to Putnam while Americans have become wealthier, their sense of community has withered (2000). Putnam has concluded that “the technology is “individualizing” people’s leisure time via television and internet” and, as a consequence, destroys social capital (2000). However, as researchers (Welman et al. 2001, Resnick, 2001) point out that only the traditional forms of social capital are eroding and being replaced by new ones. Bowling and other face-to-face local activities may have been so popular as a result of limited choice – it was too expensive or even impossible for individuals to communicate and cooperate without physically meeting each other. With the rise of online social networks and Internet the number of choices available to each individual increased dramatically. From bowling to soccer league, for example. The costs of participation in global communities has fallen to nearly 0 for individuals therefore may choose to join many other groups that share their beliefs – environmental protection, human rights etc. Moreover, while the groups

can exist in “real” world there is a fraction of groups that exist only virtually on the internet. However, these new forms may not have the same value as more traditional forms (OECD Insights: Human Capital 2007). There are also critics of social capital as a concept mainly for being too broadly and poorly defined (Portes 1998, Fine 2010). Nevertheless, the relationship among people have a value, therefore online social networks which allow to manage those relationships in a more efficient way create value for a society.

Network effects can negatively affect the value of a network. Recently, online social networks have been involved in major political scandals related to large-scale social manipulations (namely, US presidential elections and Catalan referendum for independence in 2017). Stella, Ferrara and De Domenico (2018) have analyzed the tweets and have come to conclusion that a third party botnet has been bombarding with aggressive tweets the two polarized groups of voters.

Control and prevention of automated large-scale manipulations in online social networks not only increases the costs of maintenance of a network but also the risk of sanctions from public authorities of different countries – companies can be fined or even banned and lose right to operate in countries or whole regions. On the other hand, such large-scale manipulations demonstrate the marketing and advertising potential of social networks for advertisers which could positively affect revenues of online social network companies.

Network effects can also have controversial influence on the network participants. In some configurations of online network business models profit-maximization strategies of firms may discriminate part of the network members. For example, the optimal design of pricing and monetization strategies on a competitive market (which consists of job searchers, employers and matchmakers) depends on the balance of negative network externalities between job searchers and employers (Kurucu 2008). Another example – monopolists on the market for vaccines may be incentivized to strategically leave poorer individuals without vaccines in order to increase their sales to the richer ones (Kessing and Nuscheler 2006). In this case the network effect of influence of the poorer on the propensity to consume the good by the rich creates incentives which are suboptimal for public health.

Generally online social networks should have just enough connections in order to represent the real-life relationships among their participants. There are many algorithms which rely on the assumption that all connections in online social network correspond to real social connections of an individual and allow to estimate his position in the social network (Zhou, Zhou and Zhou 2015), interests (Zhu and Lunt 2008), ranking of content (Johanson 2009), influence potential (Kim and Keng 2015). However, online social networks may not correspond to the real ones. For example, on LinkedIn many jobseekers are motivated to add more and more connections without really knowing people because these jobseekers might think that having many connections is a positive

factor in LinkedIn search algorithm and they would appear higher for headhunters who are looking for candidates. As a result, the search algorithms become less efficient and online social network companies have to spend more resources to extract value from their network – development of link quality evaluation algorithms. Some online social networks choose to set a limit of maximum possible number of connections for an individual. For example, maximum amount of friends one can add on Facebook is limited to 5000 and even though some users claim this is not enough for them, considering, for instance, the social brain hypothesis this limitation appears to be generous. According to Dunbar (1998) the quantity of surrounding people with whom each individual can have meaningful social interactions is limited to 150 (Dunbar's number) by design of our brain. Dunbar has calculated this number by extrapolation (based on the mass of the brain) of his conclusions he arrived to while observing social grooming among primates (1998). Later estimates of Dunbar's number varied from 100 to 300. (Leskovec et al. 2009, Gonçalves, Perra and Vespignani 2011, Zhao et al. 2014)

The real benefits from network effects tend to be lower than predicted by network models due to various restrictions brought by social context. For example, while mathematical models predict (Barabási 2016) and the empirical works suggest (Milgram 1967, Travers and Milgram 1969, Pool and Kochen 1978, Albert, Jeong and Barabási 1999, Lawrence and Giles 1999) that we live in a “small world” these models are too simple to capture all significant factors, that define the structure and functions of the network of social relationships. Kleinfeld (2002) has analyzed the famous Milgram's experiment papers and points out strong selection biases in favor of the existence of the “small world”. It is highly probable that Milgram (1969) was biased towards highly socially connected and prosperous individuals. Further Kleinfeld (2002) refers to another unpublished small-world study from the Milgram in Yale where the authors have tried to reproduce the small-world experiment on an socially unbiased sample where they controlled for the representativeness of low and middle-income individuals. The results of the experiment suggest that the lower-income individuals to be more socially isolated and the real-world social network has a distinguished communities which correspond to different social classes. Later Schnettler (2009) arrives to similar conclusions and suggests the ways to improve the design of small-world experiments. The computations of average network distances in Facebook (Edunov, 2016) are favoring the supposition according to which we live in a “small world”, but since the company has done the research on its internal data and only provide a brief description of the algorithm which was used for computations it is hard to assess the real weight of these findings. In summary, the studies of “small world” phenomenon are controversial and this field still has a potential for discoveries. Network companies have to ensure the behaviour of network participants does not overload the network by spamming and using bots and does not put the company at risk of law suits which may

be induced by inappropriate content which could be distributed through the network. For this reason network companies have to monitor the behaviour of its users. The cheapest and scalable options are automatic monitoring and user feedback-based systems. However automatic natural language processing and video/image recognition algorithms are only suitable for solving simple classification problems. User feedback-based content moderation requires highly engaged and motivated users (like in Wikipedia). The final decision in many cases still requires human involvement. For example, Facebook employs 30 000 content reviewers and each of them has to make up to 400 classification decisions whether each piece of content they review corresponds to the policies of the Facebook. In order to keep the margins low network companies heavily invest in improvement of automatic reviewing systems.

The growth of the user-base of any network company can be viewed as a spreading process in the network of real-world social relationships among people. Therefore if one can create a model of the real-world networks then the number of users of a network company at any point in time in the future may be forecasted using the model, information and a set of assumptions main of which are transferability and recovery coefficients and the quantity of the individuals who are "susceptible" to becoming a user as a result of interaction with others, who are already online ("Infected" ones). These are the only ingredients one needs to run simulations which would produce the distribution of possible number of users the online network company will serve at any point in time in the future. However, users do not always convert into profits. The extraction rate fully depends on the business model the company implements. The most common business models and their role in conversion of users to revenues for a network company will be classified and analyzed in the following section of this chapter.

1.6 Business models of online digital network companies

Business models of digital network companies are highly flexible. This section is devoted to general features of common online network business models. The analysis will be conducted using "Business model canvas" framework and business model patterns introduced by Osterwalder and Pigneur (2010).

As can be seen from business model canvas of different digital companies all of them have similar key activities, key partners, key resources and costs (Figures 33-36). Key activities for digital network companies always include digital platform management and may include growth of the user base and management of the services the company provides through the platform. However, both user base growth and additional services can be outsourced to key partners or bought from the market.

The key resources each digital network company has to have are the platform itself, algorithms the company uses to create value for the customers and patents for the platform and algorithm

solutions which are used to sustain competitive advantage and prevent other companies from exploitation of more efficient algorithms without investing in R&D. While the physical infrastructure and supporting software are necessary to deliver the services they are not critical resources for a digital network company and the decisions on whether to insource or outsource them is up to the company management and depends heavily on the context.

Common cost structure for digital business models includes the expenses for running the platform (hardware, software, internet traffic etc.) and employee compensation – qualified teams of developers are crucial for supporting and developing efficient and reliable high-load infrastructure and state-of-the-art algorithms.

First basic business model pattern for an online network company is “freemium” (Osterwalder and Pigneur 2010). Under such conditions the company is likely to have two customer segments. First segment of users has access to the basic set of services and features in exchange for being an active user on the network and providing user-data. The second segment consists of paid customers (one-time or occasional donators or periodical subscribers). Paid customers have access to additional paid functionality and services.

In freemium the company has to focus on three main activities: converting existing users into subscribers, acquiring new users, and retaining current subscribers. Success or failure in any of the three translates into changes in company’s revenues directly and may be magnified by viral and cascading network effects. The power of network effects varies depending on the configuration of value propositions for free users and subscribers, role of network connections and network structure. General features of freemium business model are mapped using business model canvas on Figure 33.

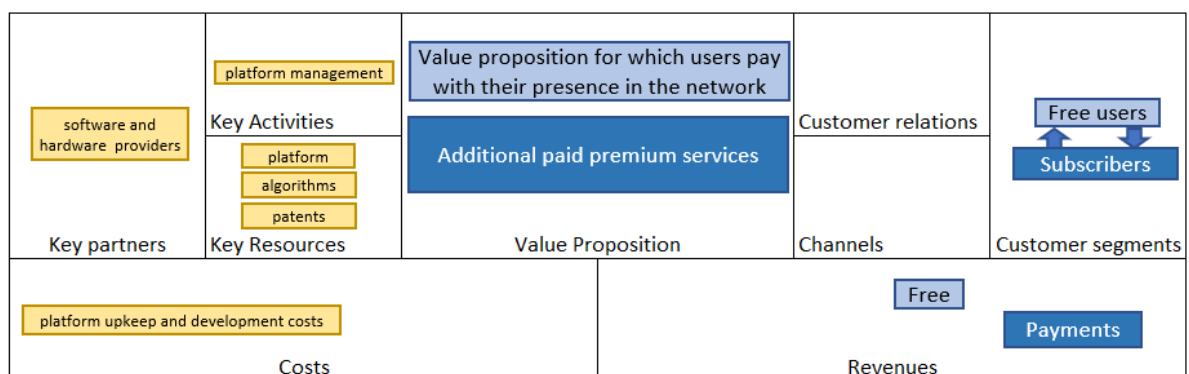


Figure 33. Business model canvas - Freemium business model

In freemium the purchases of users directly translate into revenues of companies and interests of both of them are aligned. An example of a working freemium business is XING B2C segment, which represents half of the total revenues of the company (XING SE 2019).

The second standard pattern for a digital network company is monetization through advertising. A typical advertising digital network company is Facebook and its business model is mapped on the

Figure 34. Transaction revenues of Facebook make up a tiny portion of the total revenues of the company and therefore are not included in the analysis.

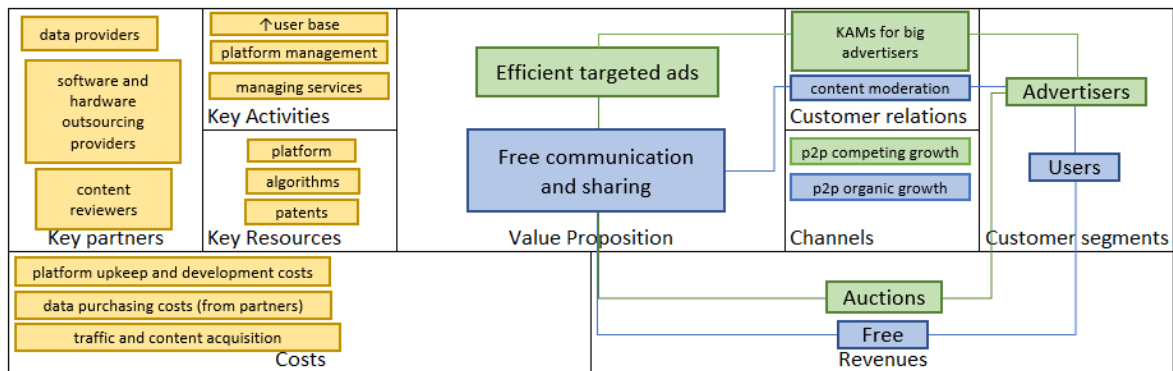


Figure 34. Business model canvas – Advertising business model of Facebook

In this model all users have access to all features of the platform for free and advertisers generate revenues for the company through purchasing the attention of the users through auctions. In this configuration the company becomes “attention merchant” (Wu 2016) and the auction price-setting mechanics maximizes its profitability – every ad view of a user is sold to the advertiser who is willing to make the highest bid. The key revenue drivers are number of users and their engagement both of which translate into increase of the number of advertisers and the bidding activity among them. Information which Facebook derives from user-data is processed in order to increase the relevance of ads shown to each user and make advertisers be willing to pay more for each ad view. From 2013 to the beginning of 2018 Facebook used to purchase data about offline activities of its users from companies like Datalogix (Lunden 2013, Hatmaker 2018). At the same time the business model of Facebook follows the “Long-tail” and “Multi-sided platform” patterns (Osterwalder and Pigneur 2010). Online advertising has expanded the advertising market size by allowing small businesses to run small-scale ad campaigns. At the same time Facebook has attempted to act like a multi-sided platform where app developers can monetize their Facebook applications through user in-app purchases. For Facebook the success of Facebook games and apps would also drive the engagement which would also result in increase of advertising revenues. Moreover, certain Facebook apps can be used to collect information about users and use it for marketing purposes. In the notorious example of Cambridge Analytica, Facebook applications in which users filled questionnaires and passed psychological tests were used for political ad targeting (Cadwalladr and Graham-Harrison 2018).

Even though Facebook has the biggest user-base in the world its monopoly power is limited. While no other online social network can compare in terms of size, Facebook constantly faces competition from smaller and local companies which focus on different aspects of online social network functionality. There are numerous alternatives for each Facebook user for both for communicating and receiving content online. Advertisers also have a variety of other channels

where they can try to reach their customers online. Both advertisers and users are price-takers without any bargaining power, but they always have alternatives for Facebook. In this situation Facebook is interested in maintaining information asymmetry about the ad price formation processes and algorithms so that advertisers pay higher bids to attract users. Unlike traditional advertising, in online advertising the relationships of the advertiser and platform are mostly based on trust. In this context traditional advertising channels like banners on websites, billboards or even newspapers have an advantage – when an advertiser purchases the placement of an ad he/she can directly make sure that the placement took place. On the contrary, when an advertiser purchases online ad there are only two directly observable characteristics of the transaction: the amount of money the advertiser paid for the ad campaign and the number of sales triggered by those ads. Facebook and other online platforms provide statistics about the campaign, but there is a risk that the numbers given by Facebook may be flawed by bugs in the algorithms or by fraudulent behaviour of part of the user-base. To maintain the trust of the advertisers Facebook periodically reports deleting bot-nets and click-farms and contentiously improves the algorithms of bot-detection. However, in the current business model setup and given the monopolistic position of the company Facebook is motivated to maintain the minimum necessary level of disclosure of information about the actual status of its user-base and minimum necessary activities to demonstrate its efforts to keep the auction for the attention of the real users fair. For a Facebook shareholder it is highly desirable to see either a higher level of disclosure or that the board of directors balances this motivation for fraudulent behaviour of Facebook management which is brought by the business model design.

The engagement of users on Facebook is driven by two types of user activity: communication and content consumption. In both of the fields Facebook faces severe competition: there are dozens of other communication and content delivery apps and platforms on the market. What makes Facebook unique is its dominant position in terms of reach – in most countries of the western world the user count is close to the total population of those countries and the total number of users makes ~35% of the population of the planet. This means that for a randomly chosen pair of individuals the Facebook will happen to be the platform one or both of them are registered on. Consequently, when these individuals would want to establish a connection online the highest probable platform of choice would be Facebook just because one or both of them are already registered and only have to find the other one. As for the content consumption – Facebook mostly relies on user-generated content. Billions and pages of users make hundreds of billions posts every year and Facebook's goal is to develop such algorithms of content distribution which would maximize the engagement. Higher engagement is mostly dependant on two factors: the time each user allocates to Facebook daily and the level of user satisfaction – in general the longer a user stays online and

the happier he is – the higher is the probability that the user interacts with marketing materials of the advertisers. For this reason all social network companies including Facebook focus on the content flow control algorithms. In addition, the content consumption profile gives a better picture about the personality of each user. By analysing likes of users it is possible to segment them according to their common preferences using various classification algorithms. Even though the user has not provided any additional information explicitly, the Facebook could still benefit from collecting data on user content preferences and consumption patterns. When a user sets up a profile in a social network he/she focuses only on those sides of the personality that in his/her opinion could be treated by the user's social connections as favourable. For example, on Russian social network VK there are almost 100 000 accounts (VK 2020) which indicate in their profile information that they currently are students of Moscow State University while in reality there are only about 40 000 (MSU 2020). On the contrary, analysis of the of content which a user consumes provides more valid information about user's interests and is of a value for advertisers in direct and indirect ways. The insights about users' preferences are primarily used for advertising. At the same time advertisers may decide to create a bot which would reproduce the behaviour of their typical user in order to keep track of the competition for the user's attention and adjust advertising campaigns better. Targeted ads not only show what seems to be relevant for a user but also can forecast what the user could need in the future and create demand for goods and services the user would not have purchased if he/she hadn't interacted with the advertisement.

The major disadvantage of Facebook's business model is the misalignment of interests of Facebook and its users. Users come to the platform to maintain their social relationships and stay informed – they are not interested in spending money for online games and they are not specifically looking for any goods or services. Running targeted advertising aiming at such users is very challenging and has proven to be far less efficient than advertising on Google, where users are motivated to get answers to their search queries (0,47 – 1.61 % CTR on Facebook vs 2 - 6% CTR on Google Search (Irvine 2019, Irvine 2020)).

Some of the users may include paid promotions in their own content, which means that part of the user-base competes with Facebook for the advertising revenues. Facebook has tried to mitigate this effect by changing the news feed algorithms to lower organic reach of the posts and give the priority to the ones promoted by paying to Facebook (Facebook 2013). This action had adverse effects. On one hand this might have increased the revenues of Facebook by converting part of the users to advertisers. On the other hand the fact that a user has to pay to reach the followers who already have subscribed for that user's news might have been perceived as unfair and toxic. This also may arguably decrease the overall quality of the user-generated content on the platform since given the decreased organic reach a blogger has an increased propensity to look for an alternative

content distribution platform where he/she doesn't have to pay to reach people who have already agreed to see him/her in their news feed.

Google and YouTube also operate in online advertising market but their business model is slightly different. The user-base of Google and YouTube comes to the platforms with more distinct motivation – to find information on the web (Google) or to consume content (YouTube). Figure 35 depicts the essential elements of the business models of Google and YouTube. The major difference with the Facebook's business model is redistribution of part of the advertising revenues to content creators. This model treats the most active users as a separate segment of content creators and financially motivates them to produce more engaging content more frequently. The global reach of Google and YouTube makes viable the long-tail content production pattern – even the most narrow topics become potentially profitable for content creators. The goal of the company is to maintain the positive cycle – higher user engagement leads to higher advertising revenues, higher amount of which is distributed to the content creators which drives the competition among them and increases the overall quantity and quality of the content which in turn drives the user engagement levels even higher and attracts more new users on the platform.

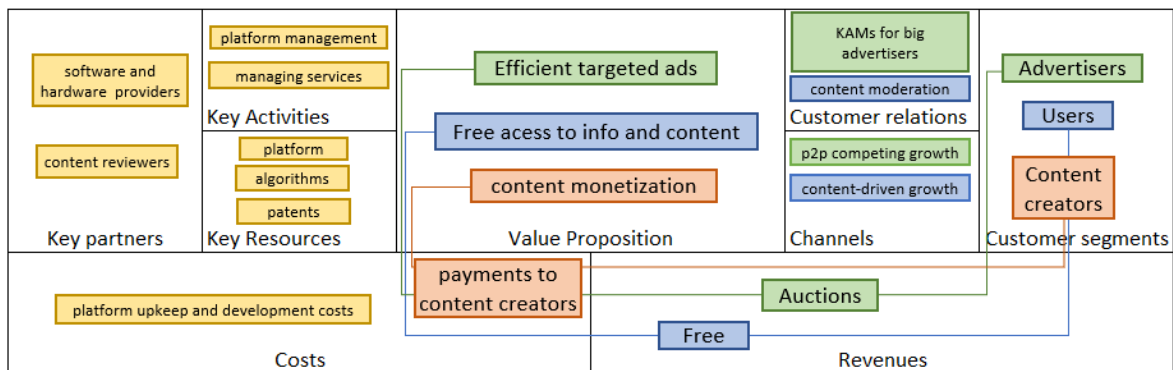


Figure 35. Business model canvas – YouTube and Google business model

Business model of LinkedIn has a more complex customer segmentation. (Figure 35) The company works in three major fields: talent solutions, marketing solutions and premium subscriptions (Microsoft 2020) and in each of the fields the company maintains a separate combination of monetization strategies. For example, in Talent Solutions LinkedIn combines freemium, advertising and long tail business model patterns. Any recruiter can perform searches on the network as a regular user and contact the suitable candidates directly, but getting a subscription he/she receives additional tools and increases the chances of finding the right candidate in a shorter time (freemium). There is also an option to make a paid job posting and promote it, so that the recruiter also becomes an advertiser. For major recruiting projects it is possible to outsource the whole process of recruitment to LinkedIn (long tail pattern). Overall customers are segmented into 5 groups and for each of them LinkedIn has a special value proposition. In order to strengthen each of its value propositions LinkedIn has acquired Bright

Media (job postings) and Lynda.com (education platform). Subscribers, corporations and advertisers generate revenues for the company while instructors receive money from the platform for the content they provide. The bulk of the customers belongs to the segment of free users and for them being on the network allows to increase their chances of building a successful career.

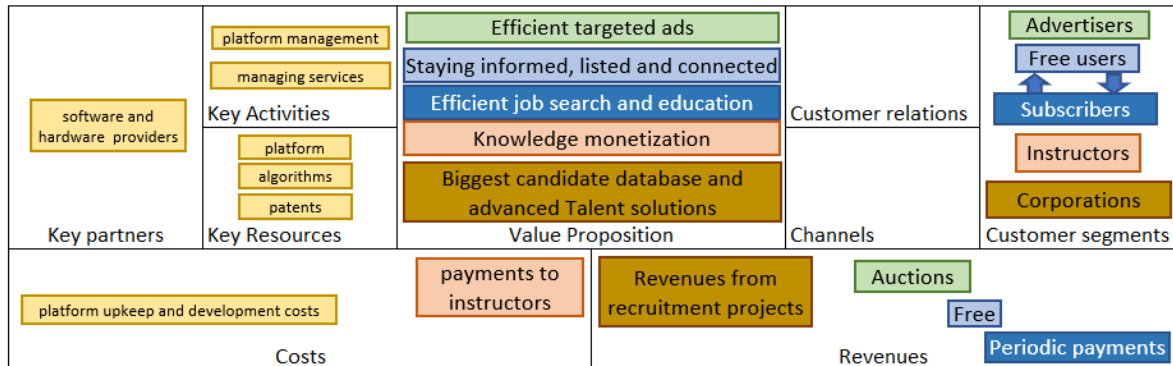


Figure 36. Business model canvas – LinkedIn business model

The life cycle of a company which rely on network effects depends mainly on the dynamics of the user-base. The user-base grows through percolation of an existing real-life social network of real people. Depending on the characteristics of the business model this process can be modelled either as SIS or as SIR epidemic processes and their modifications. In this context the goal of a network company is to cost-effectively maximize the “transmission rate” and optimize the “recovery rate” in order to get maximum LTV for each user. The number of “Susceptible” individuals is a share of population which potentially can become part of the network. The “transmission rate” characterizes the probability of a non-user to be converted into a user by another existing user through his real-life social connection. The meaning of the removal rate may vary depending on the business model. For example, for a dating website the removal rate would include both – the churn rate and the success rate of the existing users (both of them stop using the online dating service) while for Facebook the removal rate would only include the churn rate of the users.

Dynamics of the user base growth of a global online network company typically follows a 3-stage pattern. At the first stage the SIS/SIR process rapidly spreads through the existing offline social network from user to user and the user base grows exponentially. As the number of users approaches the stable level, that corresponds to a certain share of the susceptible population, which in turn depends on the network size, network structure, transmission and recovery rates, the SIR/SIS process stabilizes and the user-base dynamics starts to follow the demographical trends. In case if the ratio of transmission to removal decreases below a critical point the network will start to degenerate. If by design the failure of user’s neighbors in the network increases the chances of the removal of the user the network would be vulnerable to cascading failure events, which would make it more risky for the company which exploits the network.

In conclusion, of the business models which were scrutinized in this section rely on network effects as a result of their customer segments behaving as networks. Magnitude of those network effects and possible methods of accounting for them in valuation are examined in Chapter 2. From the business model analysis it is evident that to be more diversified and profitable online network companies aim to segment their customer base as much as they can and construct a unique value proposition (or a unique bundle of propositions) to each segment. At the same time, the flexibility of business models of all online network companies makes all of them to be potential competitors. For example, as Upwork recognizes in its 10k report, it is relatively easy for a well-established company like Google or Facebook to enter the market segment of freelance platforms (Upwork 2019). For each business model there is an optimal level of user engagement the company needs to sustain. In order to do that companies may purchase content from content providers (Netflix) or motivate creative users to fill the platform with high quality content by sharing revenues with them (YouTube). In both cases network effects apply – decisions about future content on Netflix is made based on evaluation of preferences of existing user-base and performance of the existing content and on YouTube the viral potential of content lets YouTube have higher advertising revenues.

Chapter 2. Valuation of digital network companies

2.1 Methodology

This chapter is devoted to analysis of influence of network effects on the company value and the ways to explicitly incorporate them into valuation models. Before running into the details of valuation specifics it is necessary to emphasize that this research does not take into account the fact that obtaining some of the information required to properly implement the proposed valuation models may violate policies and user agreements of the companies which are being valued as well as administrative, criminal and other laws of different jurisdictions. However, under the efficient market hypothesis the technical possibility of obtaining that information by some of the market participants means that the market will put this information into the price regardless of any related legal and ethical issues. Another disclaimer which would be reasonable to make at this point: given the complexity of businesses and network effects there is no ambition to get a true intrinsic value for any of the companies being analyzed. The main purpose of this research is to explore generic data-driven approaches to DCF valuation of a digital network company using network behavior simulation based on network science. Due to computational complexity and data-collection issues many strong assumptions will have to be made, but they will be accompanied with the ideas about how one could obtain the necessary data in order to get a better proxy (or even a real measure) for input variables and minimize the number of necessary assumption and their strength.

Basic elements of networks are nodes connected with edges therefore it is logical to start the analysis from the effect of characteristics of these basic elements on the value of networks and, subsequently, on the value of companies which administrate those networks.

The quantity of nodes and edges in a network may have adverse effects on value of a digital network company depending on the type of business and the business model. On one hand every additional node increases the value of a network by improving the network's revenue generation potential. On the other hand, every additional node in a network increases the costs of service for the digital network company. The magnitude of the increase depends on many technological and network-related factors like, for example, network density and the volume of information about each node that has to be stored and processed. In economic terms the optimal network size would correspond to several users for which marginal cost of service for a marginal user would be equal to marginal revenue. The advantage of the digital services is that the marginal costs of servicing a new client are almost always less than the revenue generation potential (and the costs tend to decrease as new technologies are being introduced therefore, other things being equal the growth of user-base should lead to increase of the company value. However, many of the digital network

companies also have offline revenue streams. For example, LinkedIn combines subscription-based revenues with offline revenues which are generated by B2B recruitment services for other companies. Even though those revenues are originally derived from LinkedIn's network of users, the relationship between them involves several random intermediate random variables – like, for example, highly unpredictable margins – the more LinkedIn goes offline the higher are the costs of scaling the business – big corporate clients, as a rule, require individual approach. Eventually the more the business model shifts in the physical space the less beneficial become the network effects of the online social network.

Connections also influence the company value. According to the Metcalfe's law "the value of a network goes up as a square of the number of its users" (Shapiro, Carl & Varian 1998). This statement is true if we assume that a) the total value of the network is the sum of values of the network for all its users, b) the value of the network for each user is a function of the number of other potentially reachable users in the network and c) each user can reach every other user in the network (which also implies that either the network has to be complete or a user can "reach" all others in two ways – directly or through longer network paths and both ways are equally costless). These are strong assumptions for physical networks, but in digital ones the costs of servicing an individual user tends to decrease further and further and the network structure serves more as a source of information for the network company rather than communication medium. This may be the reason why, for example, Zhang, Liu and Xu (2015) found that Tencent and Facebook values are in line with Metcalfe's law. However, if the Dunbar's social brain hypothesis holds, the marginal value of social interconnectedness most likely to decrease very fast after the average degree of a social network reaches 200-300 connections.

In the valuation theory Gneiser et al.(2012) have tried to bring in explicitly the value of interconnectedness into valuation. They focused on the subscription business model and assumed in their model that every user's retention rate (the probability of a XING subscriber in the year t to continue being a subscriber in the year $t+1$) depends on the interconnectedness of that user. After analysis of measures of interconnectedness the researchers found Google PR-based computation of individual retention rates to be the best choice. However, in order to calculate it they would need to have information about the whole XING network (users and their connections, dates of registration and subscription statuses). Therefore the authors have decided to use the simplest alternative available which is degree of a node (the number of connections each user has). Given the authors' assumptions about relationships between the retention rate and user degree they have chosen a calibrated arctangent function to calculate the retention rates for each user in the future periods. Gneiser et al. (2012) assume the retention rate to depend on the number of connections only for the first three years of the forecasted period and using average retention rate

for the further periods, which makes their model less dependent on the interconnectedness of users. The functional dependency of the retention rates on a node degree is assumed to be the following:

$$r_{c,i,t} = \frac{\arctan(\alpha_{t-1} * m_{c,i,t-1})}{\pi/2} \quad (2.1) \quad (\text{Gneiser et al. 2012})$$

Where $r_{c,i,t}$ is retention rate of a subscriber in a cohort c at time t , $m_{c,i,t-1}$ – degree of a node i of a cohort c at time $t-1$. The coefficient α is calculated empirically in order to fit the function to the observed retention rates for the customers during each of their first three years being subscribed in XING. Figure 37 shows the influence of the calibration coefficients on the output of the formula 2.1. The α for years 1, 2 and 3 estimated by Gneiser et al. is equal to 0.0643, 0.1560 and 0.4170. As a the result, the longer a user stays subscribed the more boost of retention he receives from each of his connections, a first-year subscriber with 100 connections, 2-nd year subscriber with 50 connections and 3-rd year subscriber with 15 connections would all have retention rates around 90% for their subsequent year. For all subsequent years the retention rate is assumed to be constant and is calculated using average growth rate of user degree (however, it is not possible to reproduce these calculations since the authors don't describe the exact way it has been done).

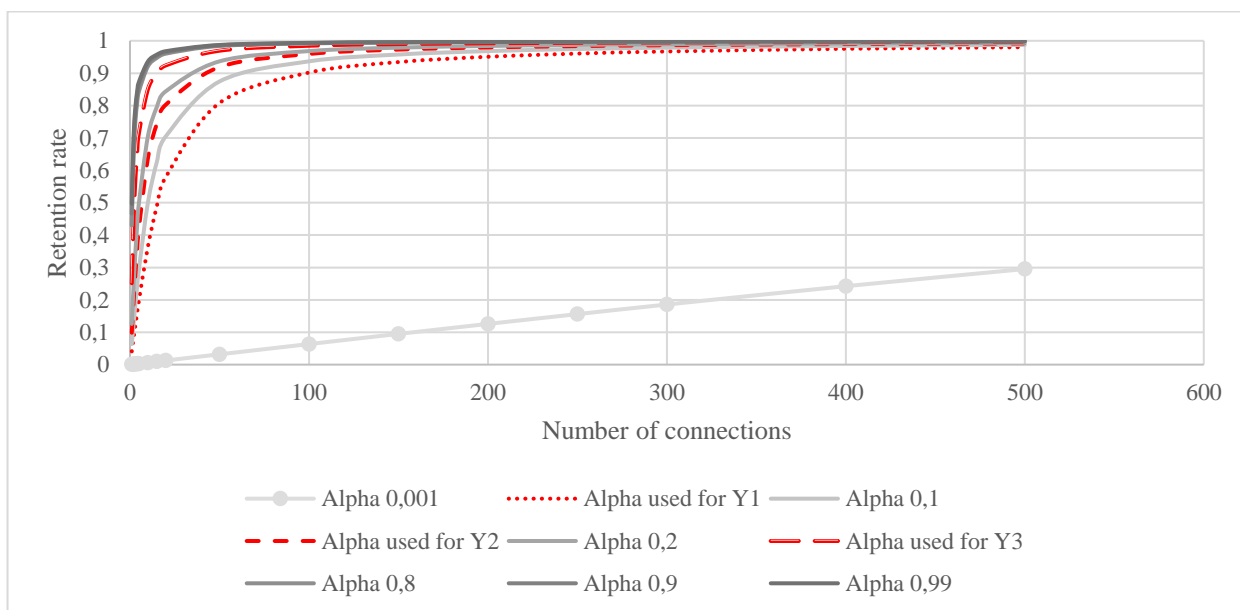


Figure 37. Modelling user-retention rate using the calibration factor α in the model of Gneiser et al. (2012)

In further calculations the researchers have computed expected cash flows from current and future users using customer lifetime value model (Koller et al. 2005; Damodaran 2002) and after making a chain of other assumptions (about margins, cost of servicing new and existing users, growth rate of the user base) they found XING to have an intrinsic value which was close to its market capitalization at the time of valuation. This research is valuable as the first attempt to formalize the relationships between network effects and company value, but the intrinsic value of XING which Gneiser et al. have obtained might not be correct because if the authors assume the retention rates for all users who stay being subscribers of XING for 3 years or more to be higher

than 97%, then it also has to mean that authors implicitly assume the expected lifetime of some of the users to be hundreds of years (for example, under this assumption 10% of users who have been subscribed on XING in 2007 will still keep being subscribers of XING 70 years later in 2077, which is clearly far from being realistic). One of the ways to account for demography in this model could be calculation of the terminal retention rate using one of the modifications of “Buy Untill You Die” approach (Schmittlein, Morrison and Colombo 1987).

More complex characteristics of networks like network density, clustering, average degree, average shortest path length and degree distribution may or may not influence the value of a network depending on the business model of a company. Over time business models evolve and the significance of all of the mentioned parameters may change as well. For example, interconnectedness probably does not influence the value of eBay too much because buyers and sellers only care about a limited number of transactions among them and do not use that network for social communications. eBay’s success relies primarily on high quality sellers and the company’s primary goal is to provide buyers with optimal price and delivery solutions for each product listed on the platform. In this case the key success factor of the company is to maintain the most geographically dense networks of high quality sellers possible for each listed product.

On the other hand for Facebook the network density of users and the geographical network density of advertisers both probably would not be primary factors which influence the value of the company. Facebook’s main goal is to keep as many real people as possible spending time in one of its products for as long as possible. In this case network density may add value through increasing the viral information spreading potential for the advertisers, but most of the ads don’t go viral. They are distributed to users who are deemed most likely to respond to the ad by the Facebook’s algorithm. If the algorithm takes into account users’ connections, then for this aspect of the business model the network density as a number becomes irrelevant. Instead, the value of interconnectedness for ad targeting depends on whether it is possible to make any judgements about user’s preferences based on his friends’ preferences, for example. In this situation having a sparse network of users with meaningful connections among them is much more valuable than having an over-connected noisy graph in which everyone randomly connects to each other. For Facebook’s business model the same logic applies to community structure, average path length and all other network characteristics.

The combination of network density and clustering defines the behavior of SIR and SIS processes on networks. It is possible to compare values of the “viral” component of different online social networks using simulations. Figures 38-43 illustrate how the spreading SIR and SIS processes, which have been observed on random networks (Figures 21-24) would behave on real ones (Facebook, Twitter and LifeJournal). The differences in susceptibility to the spreading process can

be mainly explained by differences in structures of the networks. Facebook sample has the highest clustering coefficient (which means that the graph is more dense) and lowest path length (Twitter’s average path length is more than 2 times greater). As a result, spreading on the Facebook graph occurs faster and the state of nodes can be accurately predicted. During the simulations for simplicity it is assumed that all Twitter’s and LifeJournal’s edges are symmetrical like in Facebook. The simulations of the same process on Twitter and LifeJournal graphs produce have a wider range of possible outcomes, which means that a spreading process would be less stable on them – for LifeJournal one of the simulations even predicts the spreading process to stop at time 20 without being able to activate even 1% of the network. In terms of value and risk this means that assuming the number of users being equal an advertiser would prefer to use Facebook to LifeJournal and LifeJournal Twitter to because of the differences in sustainability and speed of the viral spreading of marketing messages. At the same time, the Facebook’s better conductivity may represent a higher risk for the company since some of the spreading processes may negatively affect Facebook’s cash flows (for example, spreading of malware). If an advertiser considers a continuous marketing campaign to support the brand (for example – Coca-Cola) then the Facebook would be a better choice as well for two reasons: firstly – even though in all three networks the viral spreading is likely to enter the endemic state the Facebook achieves it faster and the proportion of Infected to Susceptible would be the highest because of the higher density of the network.

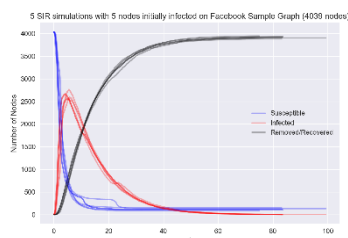


Figure 38. SIR process in Facebook

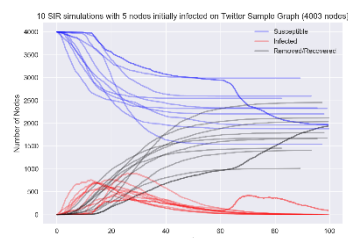


Figure 39. SIR process in Twitter

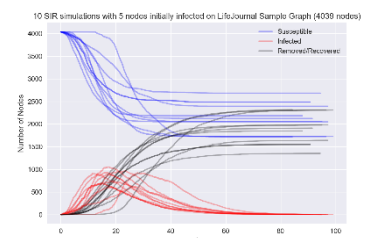


Figure 40. SIR process in LifeJournal

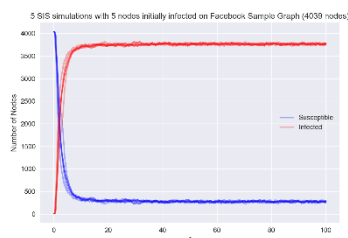


Figure 41. SIS process in Facebook

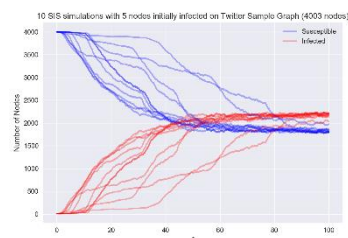


Figure 42. SIS process in Twitter

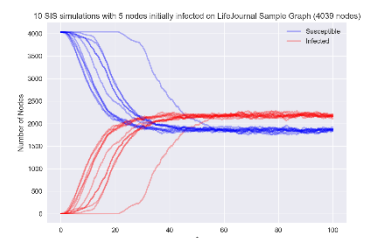


Figure 43. SIS process in LifeJournal

In the above simulations it is implicitly assumed that the Facebook’s, Twitter and LifeJournal’s users all have equal expected lifetime value if they are “activated” by a marketing campaign. The

real lifetime value of each user can be estimated, for example, by combining the analysis of user-generated content, location, device information and other data.

2.2 Valuations

All valuations are done in march 2020 and use the following inputs:

Table 3. Inputs for DCF

	US	Germany
Risk-free rate	0.65%	-0.43%
Growth rate in perpetuity	2%	2%
Equity risk premium	6.5%	6.5%
Marginal tax rate	27%	30%

2.2.1 XING

As a benchmark at first it would be reasonable to estimate the value of a network company using standard DCF approach without taking into account any network effects explicitly. Having the output of the standard model it is possible to modify part of the inputs and change part of the underlying assumptions to account for network effects which are employed by the business model. By comparing the two approaches (and interpreting the past performance of the company using network science techniques) it is possible to get a new perspective on valuation of online network companies. Following the order in which the business models were described in the previous chapter the first valuation model will be constructed for XING as it is the most successful company in the world in terms of converting free users into paid subscribers (6% of 17 mln. XING monthly active users (XING SE 2019)). However, there are 2 other segments which generate revenue streams for the company – B2B services and advertising. Even though starting from 2017 the company receives more revenues and has higher margins in them the core of XING’s business model remains its user base.

Geographically XING is focused on the German-speaking D-A-CH region with the population of about 100 million people. The company has tried to expand the geography in 2010 by acquiring companies in Spain and Turkey, but already the next year abandoned this enterprise and came back to be a focused on German-speaking market. For this reason the user-base growth is likely to have an upper theoretical limit which is approximately equal to the population of D-A-CH region.

Table 4 contains information about revenues in each of the 3 segments the company operates in.

Table 4. XING revenues by segment (2015-2019)

	2015	2016	2017	2018	2019
B2C	67.9	77.2	89.5	99.9	103.2
B2B	38.0	54.4	77.4	108.7	140.4
Ads	11.9	13.5	17.9	21.7	26.1

Table 5 contains EBITDA margins by segment for the last 5 years. In terms of profitability B2B and Advertising business has been showing much higher margins:

Table 5. XING EBITDA margins by segment (2015-2019)

	2015	2016	2017	2018	2019
B2C	66%	52%	48%	39%	26%
B2B	66%	65%	65%	65%	66%
Ads	19%	21%	37%	32%	36%

The revenues of the subscription segment are made of payments made by the subscribed users and the changes in revenues are defined by two factors: the ratio of conversion of existing subscribers to subscribers who terminate their subscription and the new subscriptions by the new users of XING.

Taking into account the main sources of XING's revenues to run a Monte-Carlo simulation the input parameters are using random distributions. The distribution of possible revenues is assumed to be normally distributed around the expected revenues, calculated assuming revenue growth rates specified in Table 6. Uncertainty about the second half of the forecasted period is reflected in higher standard deviations for 2025-2029 period. The current crisis is expected to lead to 20% revenue decline in 2020 (compared to 2019), but already in the coming year XING is expected to rapidly recover and the high growth would continue until 2025, after which the revenue growth starts to slowly converge towards long-term expected growth rate of 2% by the terminal year 2030.

Table 6. XING B2C segment revenue simulation parameters

	2020	2021	2022	2023	2024	2025	2026	2027	2028	2029
Growth rate	-20%	30%	20%	15%	10%	5%	4%	4%	3%	3%
Standard Deviation	5%	5%	5%	5%	5%	10%	10%	10%	10%	10%
Expected Revenues (Mln.)	83	107	129	148	163	171	178	185	191	196

Revenues in B2B and marketing segments have been showing a rapid growth in the past 5 years and in this model they are expected to keep growing at approximately the same rates during the high-growth period (2021-2026). Another assumption about these segments is going to be about the distribution of the expected revenues – unlike subscriptions both marketing and B2B service

revenues don't have an obvious upper demographical limit). For this reason the distribution of revenues in B2B and marketing segment has to be positively skewed and in this DCF simulation it is modeled using log-normal random distributions with the parameters listed in Table 7 and Table 8.

Table 7. XING B2B segment revenue simulation parameters

	2020	2021	2022	2023	2024	2025	2026	2027	2028	2029
Growth rate	-30%	30%	25%	25%	15%	10%	5%	5%	5%	5%
Standard Deviation	5%	5%	5%	5%	5%	10%	10%	10%	10%	10%
Expected Revenues (Mln.)	98	128	160	200	230	253	265	278	292	307

Table 8. XING marketing segment revenue simulation parameters

	2020	2021	2022	2023	2024	2025	2026	2027	2028	2029
Growth rate	-25%	30%	25%	25%	20%	15%	10%	5%	5%	5%
Standard Deviation	5%	5%	5%	5%	5%	10%	10%	10%	10%	10%
Expected Revenues (Mln.)	20	25	32	40	48	55	60	63	67	70

EBIT margin is modeled using asymmetrical triangular distribution with the following parameters: minimum of 5%, maximum of 25% and mean equal to 20% (average for the past 5 years) (Appendix C contains VBA code of the custom function, which was used to generate triangular distributions in Excel). Capex is estimated using sales to capital ratio equal to 3 (industry average). Cost of capital is calculated using inputs listed in Table 3 and levered beta equal to 1.13 which is calculated using global industry averages (Damodaran 2020). Growth rate in perpetuity is assumed to be triangularly distributed around 2% (+/-0.1%). Company's D/E ratio is expected to remain approximately the same around 1-5% with cost of debt equal to 0.32 (risk-free rate + default spread equal to 0.75% for AAA bonds on a developed market (Damodaran 2020). Considering the risk-free rate of -0.43%, beta of 1.35 and equity risk premium equal to 6% (Damodaran 2020) the resulting cost of capital of the company is equal to 7.535%. Figure 44 contains histogram of the resulting distribution of intrinsic value for the company obtained after running 10000 simulations. Table 9 contains descriptive statistics of the resulting distribution of the intrinsic value.

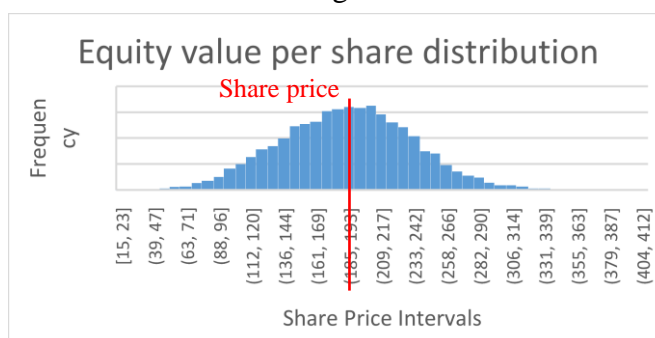


Figure 44. XING intrinsic value (EUR/share) distribution

At the time of the analysis the stock was trading around EUR 200.

Table 9. Descriptive statistics of XING's intrinsic value distribution (base case)

Mean EqV / Share	187.37
Range	14.74 – 407.64
STD	50.02
1-st quartile	152.94
Median	188.06
3-rd quartile	221.13
Skewness	0.045514
Kurtosis	0.042814

According to the model the stock is most likely to be fairly priced. The skewness of the distribution is close to zero and the range of possible values starts from EUR 14 per share up to EUR 407 per share – the probability of the stock being worth double of what it was traded at is approximately equal to the probability that it worth almost nothing.

If the uncertainty about future revenues of the company had been removed (all standard deviations were equal to 0), then the range of possible share prices would have narrowed down to 95-268 EUR per share interval. Almost half of the uncertainty about the share price in this model comes from the revenue forecast. Safer assumptions and input random variables with more narrow distributions could significantly improve the accuracy of the model. One of the ways to make safer assumptions is to rely on actual data about real-life networks and about the behavior of their users. As a proof of concept one could try to forecast the revenues from B2C segment of XING using several simplistic, but still realistic assumptions:

1. XING grows its user base through spreading from one user to another (existing users persuade other people to become new users);
2. The spreading process occurs on real-life social network through social relationships among real people;
3. Facebook friendship graph structure reflects real-life social relationships among people in DACH region;
4. The real-world social network is a scale-free network and a spreading process on real-world social network occurs at the same rate as it would occur on a small segment of the Facebook graph;
5. Only people who are between 20 and 65 y.o. are “Susceptible” to conversion into XING users;
6. People do not delete accounts on XING (recovery/removal rate = 0);
7. The probability of conversion is and the population age structure remains constant through time;

8. The ratio of subscribed users to all users every year is 98% of the previous year ratio (base year (2019) ratio = 6%);
9. The user-base of XING in 2005 was uniformly distributed across the DACH region;
10. The revenues from the B2C segment are proportional to the number of subscribed users;
11. Distribution of the quantity of “Infected” in every moment t of the forecast is normal. This assumption is made to simplify further forecasts and calculations (from the results of the previous tests of SIS and SIR models illustrated by the Figure 8 it has been shown that even though the resulting distribution does not pass strict normality tests it is still close to normal).

Appendix B contains Python code which models the quantity of XING users for the next 10 years. Firstly, the program estimates which transition rate would produce the number of “Infected” equal to the quantity of XING users in the base year (17,24 mln. in 2019). After 15 iterations of gradient descent algorithm (100 SIR simulations per iteration) it finds that on average the 0.0043 annual transition rate would have lead to 17,24 mln. users in the network by 2019. The second half of the code runs another 100 SIR simulations using the estimated transition rate. The result of the simulations is demonstrated on the Figure 45. The number of Facebook users in 2006 is used as “Initially infected” parameter of the model and it is assumed that they were evenly spread across the network.

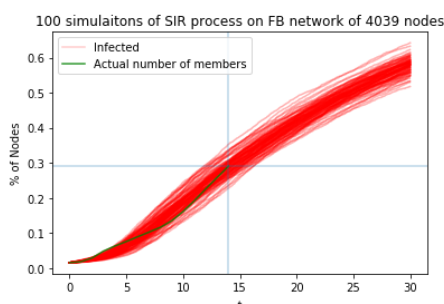


Figure 45. Forecasting quantity of XING users using SIR simulations

The revenues of B2B and Marketing segments are modeled the same way as in the base case because there are no directly observable relationships between them and the user/subscriber base. Using the mean values and standard deviations of this user-base forecast and relying on the assumptions the resulting distribution of possible share prices is characterized by the Table 10.

Table 10. Descriptive statistics of XING's intrinsic value distribution (B2C forecast using SIS)

Mean EqV / Share	152.54
Range	24 - 325
STD	49
1-st quartile	134
Median	158

3-rd quartile	226.47
Skewness	0.022866
Kurtosis	0.041102

Compared to the base case valuation (Table 9) the range of possible intrinsic values became narrower and the company starts to look undervalued.

This model can be easily modified to exploit customer lifetime value economic model used, for example, by Gneiser et al. (2012) and to reflect all other significant processes which eventually lead to extraction revenues from the user-base. One of the ways to do this in Python is, for example, to use `EoN.Gillespie_complex_contagion` function instead of simple `EoN.fast_SIR`. For example, to model additional user-base growth through traditional marketing channels it is possible to add to the SIR/SIS process random activation of nodes without directly interacting with infected ones with certain probability (or under certain circumstances). The function also allows to model competitive and cooperative SIR and SIS processes and to introduce “vaccination” of nodes – to reflect, for example, people who for some reason will not under any circumstances “transfer” the desire to become a XING user to others. It is also possible to incorporate specific rules of random node activation and vaccination, interactions with other competing and cooperate spreading processes on the same network. For example, if a person started using another social network in the past year the probability of both random and non-random “activation” as a XING user probably increases.

The main problem of using complex models is the necessity to make more and more assumptions as the complexity growth. Moreover, the realism of those assumptions rapidly increases, and, as a result the linear growth of the number of assumptions leads to non-linear growth of uncertainty about the outcomes of the probabilistic model. In such situation the optimal strategy would be to stick with the necessary minimum of the most realistic assumptions. However, online digital network companies willingly and unwillingly reveal huge amounts of different data about their user-base and business model which may replace part (or even all of the assumptions). For example, it is imaginable that one could collect data about the XING network of users and builds a model which can evaluate (or even predict) the probability of a user to become a subscriber in the following years based on all the publicly available data (job position, age, sector, presence on other social networks, subscription for other paid online services, estimated income, interests etc.). The outputs of such research could replace many of the assumptions made previously in this research about XING’s business model and future cash flows, growth and risks.

2.2.2 Facebook

In order to use network approaches for Facebook valuation at first the value of the company will be found using standard top-down DCF approach, in which future cash flows are estimated based on aggregated numbers and forecasting of company performance under the same assumptions about economy which have been used for XING's valuation in 2.2.1. The historical Revenues and EBIT margins for Facebook are listed in the Table 11. Unlike XING, Facebook almost fully relies on marketing segment revenues, which constitute more than 99% of total revenues.

Table 11. Facebook key financials 2013-2019

Year	2013	2014	2015	2016	2017	2018	2019
Revenues (Mln.)	7 872	12 466	17 928	27 638	40 653	55 838	70 697
EBIT margin	35.6%	40.1%	34.7%	45.0%	49.7%	44.6%	33.9%

In this valuation we assume that Facebook does not suffer 25% drop in revenues in year 0 like XING because the nature of those revenues is completely different. In fact, one could imagine that Facebook will even benefit from the situation with coronavirus because during the lockdown the daily audience of the platform might be more active than usual and advertisers could try to spend more on trying to pass the marketing messages to that audience. In addition to account for potential upside in future revenues in the simulation they are assumed to be lognormally distributed with the parameters listed in Table 12.

Table 12. Facebook revenues simulation parameters

	2020	2021	2022	2023	2024	2025	2026	2027	2028	2029
Growth rate	25%	20%	20%	15%	15%	12%	10%	5%	3%	3%
Standard deviation	5%	5%	5%	10%	10%	10%	10%	15%	15%	15%
Expected Revenues	77 930	99 685	126 860	127 120	162 905	182 808	191 887	200 821	208 286	188 803

It is assumed that the company's high growth period will last for 5-8 more years and will converge to the growth rate of the global economy. The increasing standard deviation of the estimated revenues reflects the uncertainty about the future profits and the assumption about log-normal distribution shifts the uncertainty in favor of the company leaving long right tail in the probable revenue numbers, which is reasonable given the fact that Facebook is continuously trying to develop new products which potentially could allow the company to start extracting much more revenues from its user-base directly. If the company manages to add the subscription-based data-driven services then given its huge user-base the revenue growth might be a multiple of what the company manages to deliver now. The reinvestment rate is assumed to be equal to XING's – for

each additional dollar in revenues the company on average will have to pay 33 cents. Growth rate in perpetuity is assumed to be triangularly distributed around 2% $\pm 0.1\%$. Since the company most likely has already reached the maximum possible penetration in the most economically developed regions in the world and does not have to heavily invest into user acquisitions anymore the EBIT margin is assumed to be also triangularly distributed with minimum of 25%, maximum of 40% and mean of 30%, which also will skew the resulting equity value distribution to the right. Facebook's current beta is assumed to be equal to 1.05. Using the risk-free rate of 0.65% (10-y US government bond in March 2020) and equity risk premium of 5.9% (Damodaran 2020) the resulting cost of equity is equal to 7.73%. Facebook's optimal D/E ratio is 15% and the company is expected to converge towards the target debt ratio and maintain it in the perpetuity. Using the levered beta of 1.2 and the pre-tax cost of debt equal to 1.4% (risk-free rate + default spread for AAA corporate bonds equal to 0.75% (Damodaran 2020)) the resulting WACC is equal to 6.8%.

Under these assumptions after running a Monte-Carlo simulation one could obtain the equity value distribution pictured on Figure 46.

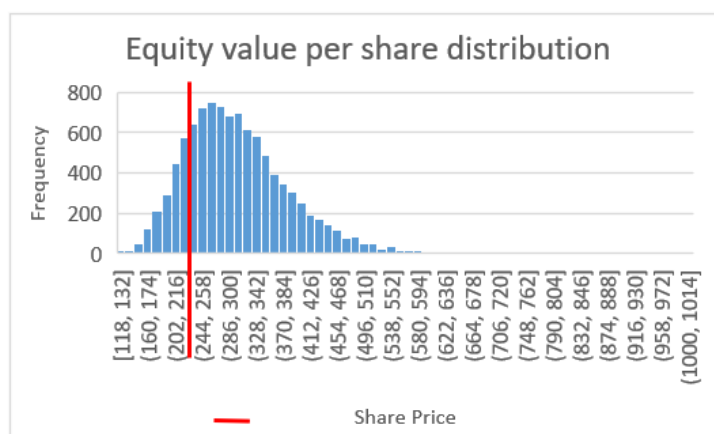


Figure 46. Facebook intrinsic value distribution (base)

From the distribution it can be concluded that the company is most likely to be undervalued. The characteristics of the resulting distribution are listed in Table 13.

Table 13. Descriptive statistics of Facebook's intrinsic value distribution (base case)

Mean EqV / Share	306.33
Range	117.9 – 1001.4
STD	84.96
1-st quartile	246.41
Median	294.36
3-rd quartile	351.60
Skewness	1.1451
Kurtosis	3.2039

The resulting mean intrinsic value per share exceeds the current market price by more than 30%. Under some scenarios the model predicts that the equity value per share could be even 1000, which is 5 times greater than current market price.

As a result of the standard top-down valuation the distribution of possible intrinsic values ranges between 117 and 1000 USD. Further it will be tested whether the network modelling could decrease the uncertainty and make some of the previous assumptions unnecessary.

In order to use the same model which was used for valuing XING's users in addition to previously made assumptions it is necessary to make a number of new ones:

1. The spreading of Facebook occurs in all countries of the world simultaneously (except for China);
2. Only people in the age group between 15 and 65 years old are susceptible;
3. The rate of transmission is the same for the whole susceptible population;
4. The recovery/removal rate is equal to zero – people don't quit using Facebook and the number of deceased users is entirely covered by the number of those who reach the age of 15;
5. The value of users for marketing purposes is homogeneous among the countries;
6. Other local online social networks don't affect the spreading of Facebook;
7. The total population remains constant (equal to the average of the period – around 7 Bln.);
8. The revenues of the company depend on two components: the growth rate of the user-base and the average revenues per user:
9. Growth rate of the user-base is a SIR-type process with constant transmission and recovery rates;
10. Average revenues per user in will grow with the same CAGR as they did during the last five years and linearly converge to 0 by the terminal year.

Under the above mentioned assumptions the resulting distribution of price per share would look similar to Figure 47.

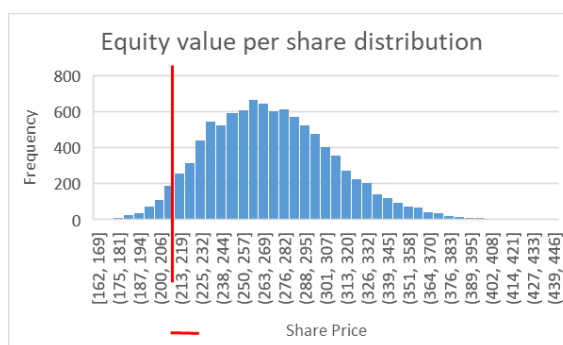


Figure 47. Facebook intrinsic value distribution (SIR Simulation-based)

Figure 48 depicts the dynamics of 100 SIR simulations using the spreading rate computed by gradient descent. Facebook’s computed spreading rate turned out to be 0.01937 which is almost 5 times higher than it was estimated for XING. The green line represents the number of users reported by the company.

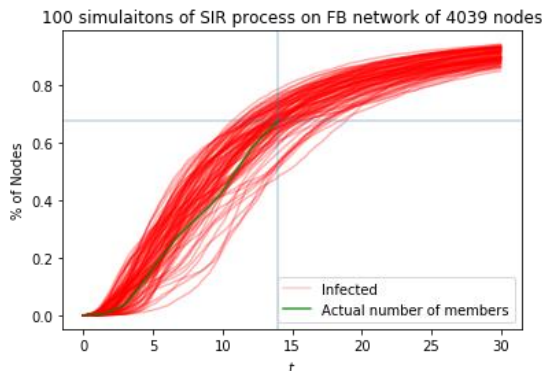


Figure 48. Forecasting quantity of Facebook users with SIR simulations

As it can be seen from the Table 14, using SIR model and making additional assumptions significantly narrowed the range of probable intrinsic values from 117.9 – 1001.4 to 162.3 - 440.79 and decreased the standard deviation by 2 times.

Table 14. Descriptive statistics of XING's intrinsic value distribution (B2C forecast using SIS)

Mean EqV / Share	271.79
Range	162.3 - 440.79
STD	38.95
1-st quartile	243.83
Median	268.86
3-rd quartile	296.10
Skewness	0.504239
Kurtosis	0.409667

Almost all of the additional assumptions which have been made in this SIR-based valuation were highly judgemental. The following section of this research is devoted to finding ways to replace the unnecessary assumptions and judgements by using data analysis.

2.2.3 Adapting SIR and SIS models for more accurate forecasting using data

The model which have been used in the previous sections is highly abstract and has many simplistic assumptions. This section focuses on the algorithms of collecting real-world data in order to make some of the previously made assumptions unnecessary.

The first basic assumption was about all users to be equally valuable for the network company. In the real world this would obviously be a far-fetched assumption to make. There is tremendous economic inequality between different regions of the world. For the purposes of valuation, however, it is possible to use global statistical data in order to try to estimate more precisely the market size. During the last decade bottom-up user valuation approaches have also become feasible. The value of each user can be estimated by analyzing data which the user and social network provide. Online social networks have to keep the data about their users available due to their business models. Even though the companies are trying to restrict the data collection from their networks it will always remain possible to download at least a significant sample of data. The following proxies can be used in order to estimate the value of each individual user:

1. The average value of other user in the neighborhood,
2. Demographics, geography and language.
3. Historical preferences of content consumption and interaction.
4. Travelling activity.
5. User-generated content,
6. Combination of all of the above. For example, it is possible to detect objects on the pictures which user publishes. If user has published a picture with a car it is possible to automatically recognize the type of the car and estimate its market price directly from the picture. The second step for the algorithm would be to detect whether the car belongs to the user who has published the picture. This can be done by combining, for example, information about the car position on the picture (whether it is on or above the ground) and the information about other users who have taken pictures in this location – if many people take a picture of the same car in the same place then it does not probably belong to any of them.

The second assumption has been made about the constant network structure, but in reality over time the network structure is likely to change. It is also technically possible for someone outside of a network company (for example, Facebook) to collect information about the whole network structure. For instance, Social Data Hub CEO in 2014 claimed that they have full copies of the Russian segment of Facebook for each month since 2010 (Hachuyan 2017).

It is also possible to account for uneven network transitivity for each particular type of SIR/SIS process by constructing models with varying edge weights which would be based on real-world data about the users and their preferences and real world social network structures.

In order to account for demographic factors it is possible to use demographic forecasts and incorporate them into the simulations of network behavior. Different gender and age might also

affect the properties of the networks and it is also possible to reflect different behaviors in the simulation models.

Both for Facebook and XING it has been assumed that they spread through the real-life social network without any competition. In reality, XING has always had a strong competitor – LinkedIn, and Facebook had and in some countries still has local competitors. To account for this factor it is possible to amend the SIR and SIS simulation with the second spreading process which would influence, for example, the probability of spreading of XING usage in Germany. And, on the contrary, sometimes several different SIR and SIS processes amplify each other. In business environment these effects often attribute to terms “ecosystem” and “platform”. For example, different SIR and SIS type processes in Google’s ecosystem would always be designed in order to amplify each other – if the user converts into Android user he is most likely to use the default applications which are specifically designed and optimized for his particular type of device. However, for someone outside of Google it is probably not possible to collect enough data about the behavior of its users in order to construct more fine-tuned data-driven valuation models.

For a proper valuation of companies which rely on complex business models with more than two types of users a proper model would require data about all parties involved, and it is also possible to collect it in some cases. For example, in order to estimate the competition in Google Ads or Yandex (Russian search engine) it is possible to establish several accounts and to simply use the advertising budget forecast tool to collect data about both the traffic and the pricing for all commercial search queries. By repeating the procedure on a weekly basis it is probably possible to construct a model which could forecast the advertising revenues of Google and Yandex.

For Facebook a proper valuation model would involve both users and advertisers and the main goal of simulation would be to forecast the advertising activity for most of the major advertisers. In terms of modelling this could be described by a bipartite graph with two types of nodes – users and advertisers and the simulation would attempt to forecast the possible outcomes of various interactions among them. Moreover, Facebook is probably too global to be able to incorporate all country and cultural differences into one simulation model (for example, average revenue per user in the US is 15 times higher than in Asia-Pacific). In this situation for a better valuation it would be necessary to collect data and run different simulations for each of the regions the company operates in.

LinkedIn’s data-driven valuation model based on network science would incorporate all of the features described above because in its business model the company attempts to combine B2B services with marketing, B2C subscriptions and YouTube-like content monetization features.

Chapter 3. Empirical analysis of the quality of annual reports

3.1 Similarity analysis of annual reports

One of the main issues related to the “number of users” metrics is the inability of companies to compute them accurately. In their disclaimers Facebook, LinkedIn, XING state that all the numbers related to users in the report are best estimates based on internal company methodology. One of the ways to estimate the risk of the number of users to be substantially different from the real one is to compare the evolution of the annual reports of each company as well as compare legal disclaimers’ similarity across companies which administrate online digital networks. In this chapter the textual analysis is performed using natural language processing algorithms (NLTK project 2020). Annual reports will be compared using cosine similarity algorithm, which will transform each text into vector and, as a result, will evaluate their similarity between 0 and 1.

The companies selected for analysis are listed in the Table 15.

Table 15. Company reports selected for textual analysis

Company	Ticker	Annual 10-k reports
Amazon	AMZN	1997-2019
Activision Blizzard	ATVI	1995-2019
Ebay	EBAY	1998-2019
Facebook	FB	2012-2019
Google	GOOGL	2004-2019
LinkedIn	LNKD	2011-2015
Mail.ru	MAIL	2010-2019
Microsoft	MSFT	1994-2019
Match Group	MTCH	2016-2019
Snapchat	SNAP	2017-2019
Twitter	TWTR	2013-2019
XING	XING	2006-2019
Yandex	YNDX	2012-2019
Royal Dutch Shell	RDSA	2005-2019
Exxon Mobil	XOM	1993-2019

In the sample there are 13 companies business model of which relies on online platforms and social networks (fully or partially) and have an established network of users. In order to estimate the portion of similarity which comes from the 10-k format itself and all of the legal requirements the reports of 2 oil companies have been included in the sample – Royal Dutch Shell and Exxon Mobil.

Firstly each company reports have been compared to each of their past and future versions. The resulting similarity matrixes can be viewed in Appendix D. The Python code used to produce the similarity matrixes is listed in the Appendix F. After the year 2000 year over year similarity coefficient remains above 0.9 for the majority of the reports of all companies. However, it is possible to observe different periods when the reports stayed approximately the same - for Amazon it was 2000-2002, 2003-2011, 2012-2016 and 2017-2019. Activision Blizzard, Ebay, Microsoft, XING and ExxonMobil show rapid changes in their reports during the first 5-10 years (period between 1995 and 2002), however these can be the results of changes of format of SEC's Edgar database and not of the company reports themselves. Overall, the year over year testing shows that linguistically all of the company reports maintain the same structure and wording during the past 20 years.

The next test is devoted to evaluation of the similarity across the annual reports for each year. The corresponding matrixes are calculated using the same algorithm as in the previous test, but this time the reports of each company are compared to the other companies' reports for each year. The resulting cosine similarity matrixes are listed in the Appendix E. Python code to produce them can be found in the Appendix F. The similarity matrixes show that a) the reports of oil companies (Shell and ExxonMobil) are significantly "less similar" to the reports of digital companies. The average similarity coefficients by years are listed in the Table 16. The similarity coefficients of the reports of almost all of US-based digital companies are around 0.6.

Table 16. Average report similarity coefficients by years

Year	AMZN	ATVI	EBAY	FB	GOOGL	MAIL	MSFT	MTCH	RDSA	SNAP	TWTR	XING	XOM	YNDX
1994							0,71						0,71	
1995		0,65					0,63						0,61	
1996		0,66					0,62						0,59	
1997	0,52	0,60					0,59						0,54	
1998	0,46	0,61	0,55				0,51						0,50	
1999	0,57	0,57	0,57				0,55						0,57	
2000	0,58	0,57	0,44				0,47						0,50	
2001	0,61	0,51	0,59				0,55						0,54	
2002	0,64	0,55	0,59				0,56						0,56	
2003	0,69	0,67	0,63				0,66						0,62	
2004	0,67	0,62	0,60		0,63		0,63						0,57	
2005	0,61	0,55	0,53		0,55		0,57		0,46				0,54	
2006	0,58	0,52	0,51		0,52		0,55		0,45			0,46	0,52	
2007	0,56	0,49	0,50		0,48		0,53		0,41			0,30	0,51	
2008	0,60	0,52	0,53		0,53		0,55		0,46			0,30	0,53	
2009	0,60	0,53	0,53		0,50		0,56		0,46			0,30	0,53	
2010	0,62	0,54	0,51		0,58	0,48	0,56		0,50			0,42	0,53	
2011	0,63	0,57	0,51		0,61	0,51	0,57		0,51			0,44	0,50	
2012	0,61	0,52	0,47	0,57	0,57	0,48	0,52		0,44			0,38	0,46	0,40
2013	0,60	0,54	0,50	0,60	0,61	0,49	0,55		0,46		0,59	0,39	0,44	0,41
2014	0,61	0,56	0,53	0,63	0,64	0,49	0,56		0,44		0,62	0,39	0,46	0,43
2015	0,61	0,56	0,62	0,63	0,58	0,47	0,55		0,44		0,61	0,35	0,42	0,42
2016	0,62	0,58	0,62	0,63	0,61	0,47	0,56	0,53	0,41		0,62	0,35	0,48	0,42
2017	0,62	0,56	0,61	0,64	0,60	0,44	0,55	0,56	0,40	0,58	0,62	0,33	0,41	0,41
2018	0,62	0,56	0,53	0,63	0,60	0,41	0,56	0,56	0,40	0,59	0,62	0,36	0,47	0,37
2019	0,63	0,57	0,60	0,61	0,61	0,49	0,56	0,46	0,41	0,59	0,62	0,41	0,48	0,41

Companies based in Germany and Russia as well as oil companies expectedly demonstrate much lower cosine similarity coefficients (0.3-0.4) due to different reporting requirements for different countries and different sectors.

The cosine similarity coefficients for US-based listed digital network companies remain the same for the whole period – around 0.6, which means that there hasn't been convergence among the reports and the companies kept their own structure and wording.

Changes in the stocks' betas have been regressed on the cosine similarity of each report with the previous year's report. The outputs of the regression are listed in the Table 17. The results suggest that cosine similarity does not explain variance in a stock's beta and judging by similarity coefficients it is not possible for the users of the annual reports to forecast future company-specific risks based on the texts of the reports.

Table 17. Regression outputs - Changes in stocks' betas and the cosine similarity of companies' reports

Parameter	Value
R ²	0,00004287
Intercept	0,0416
Coefficient	-0,02999953
Standard error	0,397265913
T-statistics	-0,07551500
P-value	0.9399

3.2 Lexical analysis of annual reports

Firstly for the lexical analysis all most frequently used modal verbs, verbs, nouns and adjectives from all reports have been collected and the frequency distribution plots have been constructed. Out of all of the most frequently used words the ones which are commonly used in the risk context have been selected using the Python code which can be found in the Appendix F. Using the list of selected keywords, lexical distribution plots have been constructed for all years and for every company. Each plot (Figures 50-64) depicts the distribution of the selected words and the older occurrences are added to the plot using lighter colour. The keywords which were included are: “legal”, “regulatory”, “impact”, “incur”, “harm”, “risk”, “product”, “network”, “platform”, “data”, “information”, “user”, “might”, “must”, “would”, “could”, “may”. These are 2-3 top-frequently appearing keywords obtained after analysing the reports using Parts of Speech (POS) tagging to select modal verbs, nouns and adjectives.

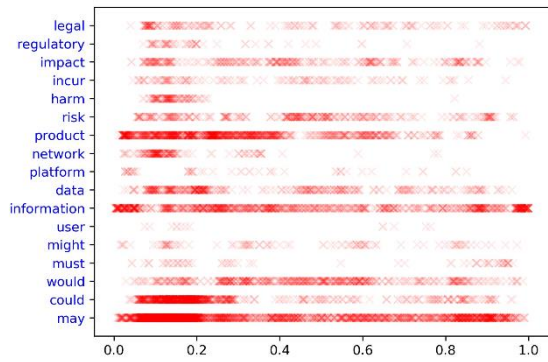


Figure 49. Lexical dispersion plot - Amazon

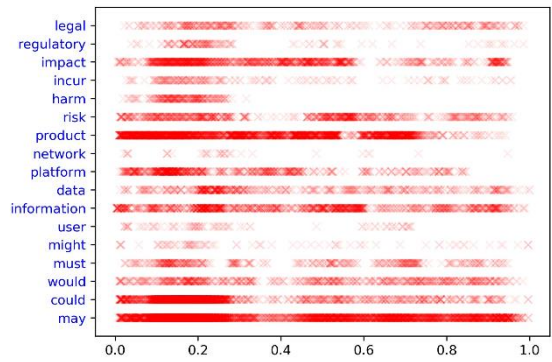


Figure 50. Lexical dispersion plot - Activision Blizzard

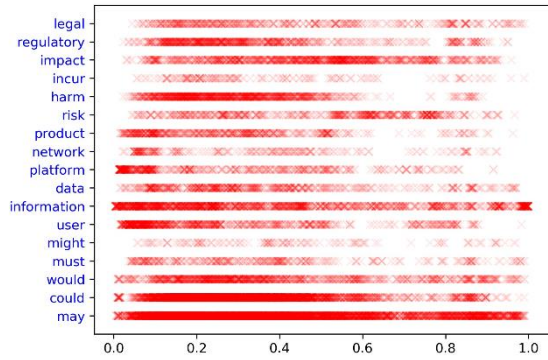


Figure 51. Lexical dispersion plot - Ebay

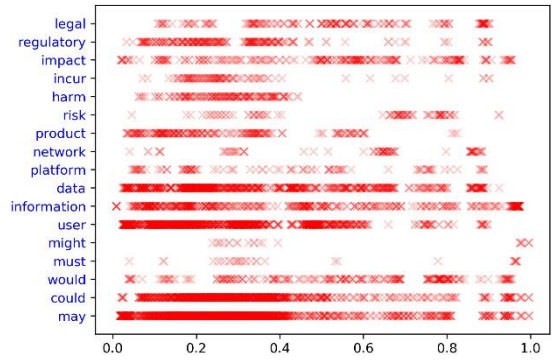


Figure 52. Lexical dispersion plot - Facebook

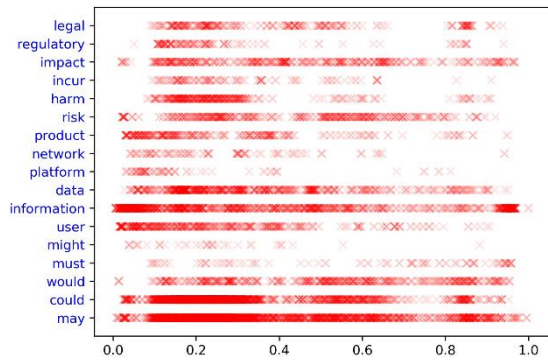


Figure 53. Lexical dispersion plot - Google

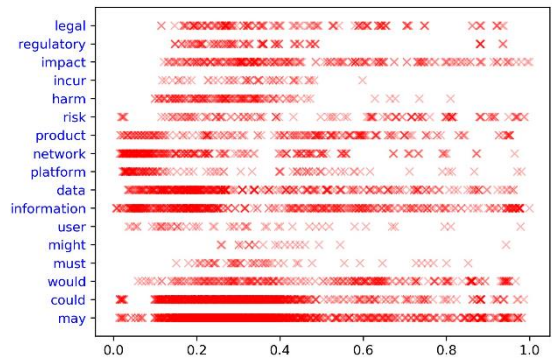


Figure 54. Lexical dispersion plot - LinkedIn

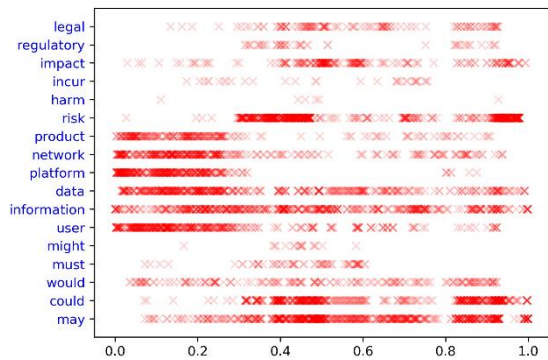


Figure 55. Lexical dispersion plot - Mail.ru group

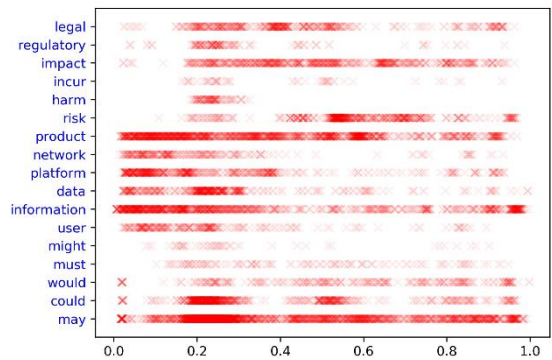


Figure 56. Lexical dispersion plot - Microsoft

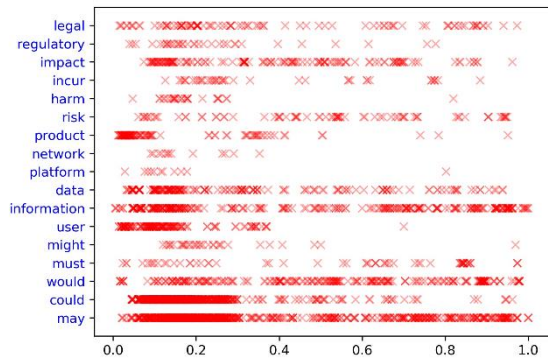


Figure 57. Lexical dispersion plot - Match group

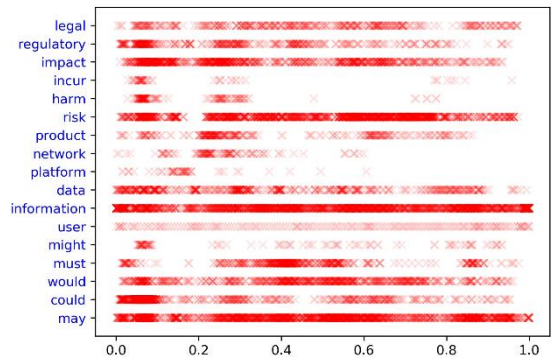


Figure 58. Lexical dispersion plot - Royal Dutch Shell

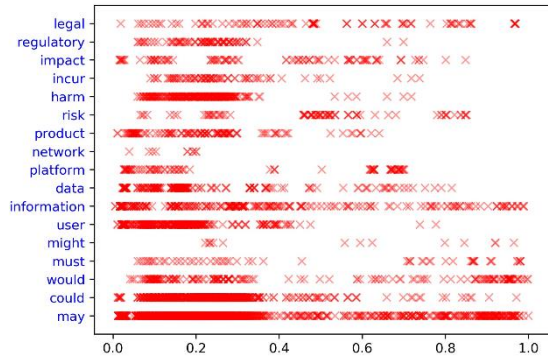


Figure 59. Lexical dispersion plot - Snapchat

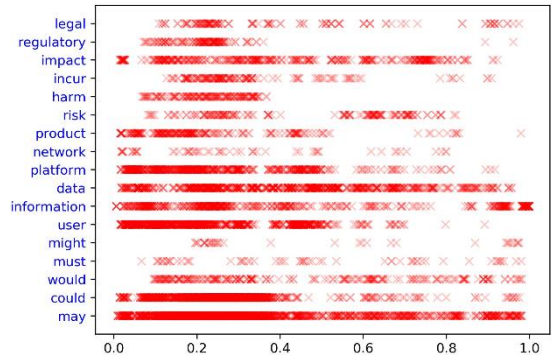


Figure 60. Lexical dispersion plot - Twitter

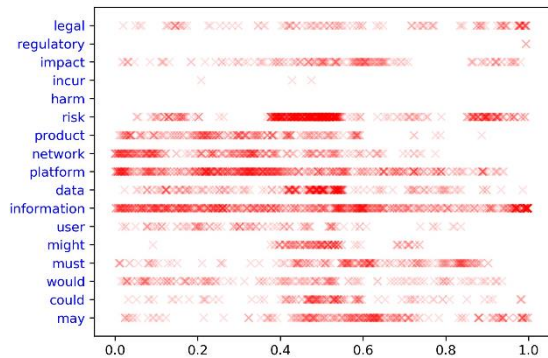


Figure 61. Lexical dispersion plot - XING

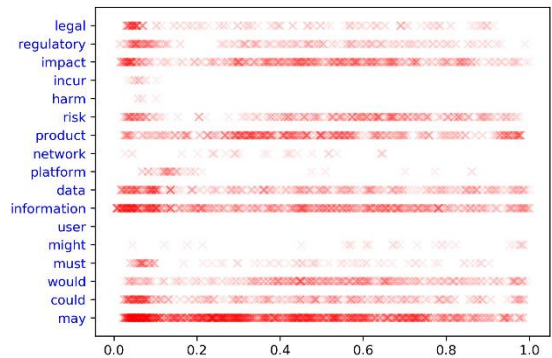


Figure 62. Lexical dispersion plot - ExxonMobil

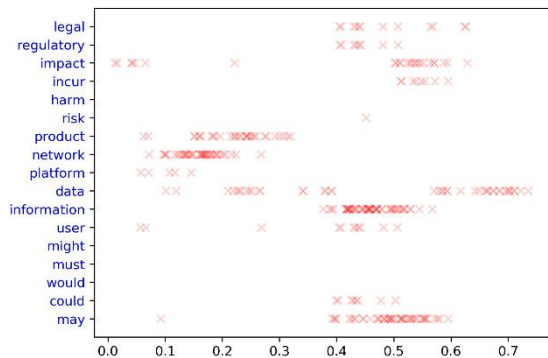


Figure 63. Lexical dispersion plot – Yandex

US-based companies have been intensively using modal verbs “may” and “could” to express uncertainty about the future all the way through their reports during the past 20 years. “would” is usually used to warn the reader about hypothetical negative scenarios which may “impact”

financial results of the company. Supposedly, as a result of the series of class action suits which have been filed by investors in the US the companies have been trying to use these verbs to decrease the risk of getting penalized for misleading their investors by any of their forward-looking statements. On the contrary, XING and Yandex focus on risks only in special sections in the middle of their reports. Mail.ru Group and LinkedIn clearly separated the topic of risk and uncertainty from their products, platforms, networks, data and users since the words mostly appear in different parts of the reports. For most of the US-based companies in the sample the keyword “risk” has moved to the first half of the report through the last 20 years. Microsoft and Yandex have the most distinctive white spots in their lexical dispersion graphs, which implies that the lexical structure of the reports stayed pretty much the same through the whole period and certain words appeared in certain parts of the reports and never appeared in the other ones.

Conclusion

It has been demonstrated how network effects could be taken into account explicitly. Basic properties of networks and processes which occur in them have been reviewed. Business models of online network companies have been described and mapped using the business model canvas. Older approaches of taking into account the users' interconnectedness (Gneiser et al. 2012) have been analyzed and improved. SIR and SIS simulation models have been used to simulate the behavior of real-world networks and have significantly narrowed the range and distribution of possible simulated intrinsic values of Facebook and XING. Although, many other strong assumptions had to be made in order to apply the simulation models it has been discussed that gathering more real-world data about the networks of interest would make many of those assumptions unnecessary. Synthesis of publicly available user-data from different online social networks can also help to construct predictive models to forecast revenues of subscription-based and marketing companies like LinkedIn, Netflix, Spotify and XING. Further data collection, studies and tests of network-based valuation models are necessary to be able to make a judgement on the efficiency of this type of valuation models. Empirical analysis of companies' reports has shown that the reporting standards in the US remained the same in the past 20 years and the changes in words and structure of the reports does not allow to predict changes in beta of the corresponding stocks in the future. US-based digital companies have presented high cosine similarity between their reports in terms of length, words being used and the structure of the reports and the reports do not change significantly with time for the last 20 years. The regression analysis of changes in beta against year over year similarities of corresponding annual reports has shown that changes in the wording and structure of the reports don't allow the users of the reports to predict the changes in company-specific risk. Lexical analysis has shown that US-based companies usually focus on the legal disclaimers and the description of the risk factors at the beginning of the reports while XING and Russia-based Mail.ru clearly separates risk and uncertainty from its products, data and business and Yandex only discloses financials. The textual analysis confirms that for online network companies the corresponding online networks themselves along with financial information would allow to construct much more accurate valuation models rather than using the annual reports alone.

Appendix A. Python code – Chapter 1

```
1.  #!/usr/bin/env python
2.  # coding: utf-8
3.
4.  # # Network effects as value drivers for online digital companies
5.
6.  # ## Preparations
7.  """
8.  # Requirements:
9.  # -versions of Python and libraries:
10. # --Python 3.6.6
11. # --EoN 1.0.8
12. # --matplotlib 3.1.1
13. # --networkx 2.4
14. # --numpy 1.17.3
15. # --pandas 0.25.2
16. # --pandas-datareader 0.8.1
17. # -Equipment:
18. # --HP Envy 15 171-nr Core i7 4700mq 12GB RAM
19. """
20. #----- sample config.txt contents:-----
21. """
22. # This file contains all variable input parameters
23. [Paths]
24.
25. # Folder with Input Datasets:
26. InDatasetsPath=C:/D/Documents/Studies/2017-
2019/9_Thesis/3_Data/in_datasets/
27.
28. # Folder for Output Datasets:
29. OutDatasetsPath=C:/D/Documents/Studies/2017-
2019/9_Thesis/3_Data/out_datasets/
30.
31. # Folder for Figures:
32. OutFiguresPath=C:/D/Documents/Studies/2017-
2019/9_Thesis/3_Data/out_figures/
33.
34. [InFileNames]
35. # Input Datasets Filenames:
36. SourceURL=https://snap.stanford.edu/data/
37. FB=facebook_combined.csv
38. LJ=soc-LiveJournal1.txt
39. TWTR=twitter_combined.csv
40. TWTR_Scalability=twitter_combined.csv
41. LJ_Scalability=soc-LiveJournal1.txt
42.
43. [OutFileNames]
44. # Output Datasets Filenames:
45.
46. [OutFigNames]
47. # Output Figure Names (degree distribution plots):
48. Path=Path Graph
49. Star=Star Graph
50. Complete=Complete Graph
51. FB=Facebook Sample Graph
52. TWTR=Twitter Sample Graph
53. TWTR_Scalability=Twitter SIR process scalability
54. LJ_Scalability=LifeJournal SIR process scalability
55. LJ=LifeJournal Sample Graph
56. Random=Random Graph
```

```

57. Barabasi=Barabasi-Albert Graph
58.
59. [Parameters]
60. # Stats computations is resource-demanding for big graphs
61. StatsOn=False
62.
63. [Parameters_Drawing]
64. # On/off drawing of Graph on top of histogram (Slow for graphs with over
1000 nodes)
65. DrawGraphOnTopOfHist=True
66.
67. [N]
68. # Desired Number of Nodes for graphs generated by the script:
69. Path=20
70. Star=20
71. Complete=20
72. Random=1000
73. Barabasi=1000
74.
75. [Sampling]
76. # Sampling parameters
77. TWTR_sample=100000
78. TWTR_random=False
79. TWTR_bigfile=True
80. TWTR_startline=1
81.
82. # SIS and SIR tests (sample size not set here - it will be iterative)
83.
84. TWTR_Scalability_random=False
85. TWTR_Scalability_bigfile=True
86. TWTR_Scalability_startline=1
87.
88. LJ_Scalability_random=True
89. LJ_Scalability_bigfile=True
90. LJ_Scalability_startline=5
91.
92. LJ_sample=5000
93. LJ_random=False
94. LJ_bigfile=True
95. LJ_startline=5
96.
97. FB_sample=4037
98. FB_random=False
99. FB_bigfile=False
100. FB_startline=1
101.
102. [Percolation_Basics]
103. # random graph connection probability:
104. p=0.05
105. # number of edges for each additional node in Barabasi-Albert random
graph:
106. barabasi_edges=1,3,5
107.
108. # percolation parameters. Percolation timing:
109. rho_default=5
110. tmax=100
111. iterations=5
112. # transmission rate:
113. tau=0.1
114. # recovery rate:
115. gamma=0.09
116. # uncomment to set custom rho (% of S initially infected):
117. #rho = 0.01

```

```

118. """
119.
120. # ---- specifying config.txt location ----
121. # In order to separate the code from variables I use configparser module
122. # Input variables are parsed config.txt.
123. # If: the file is located at C:/config.txt
124. #     no need to change anything in the code
125. # else:
126. #     If config.txt is located in another directory specify full path
    here
127. configTXTlocation = "" # (for example "C:/folder/subfolder/config.txt")
128. # import all the necessary libraries:
129. import networkx as nx
130. import statistics
131. import matplotlib.pyplot as plt
132. import collections
133. import EoN
134. import csv
135. import re
136. import pandas
137. import random
138. import configparser
139. # Turn off Warnings
140. import warnings
141. warnings.filterwarnings("ignore")
142. # --- Parsing config.txt ---
143. if configTXTlocation == "": #
144.     configTXTlocation = "C:/config.txt"
145. config = configparser.ConfigParser()
146. config.read(configTXTlocation)
147. # -- Paths
148. Path_In_Datasets = config['Paths']['InDatasetsPath']
149. Path_Out_Figs = config['Paths']['OutFiguresPath']
150. # -- Names
151. FigNames = config['OutFigNames']
152. InFileNames = config['InFileNames']
153. OutFileNames = config['OutFileNames']
154. # -- Parameters
155. Params_Drawing = config['Parameters_Drawing'] # Parameters for Function
    DrawGraph()
156. DrawSubgraph = Params_Drawing['DrawGraphOnTopOfHist'] # on/off Drawing
    Graph (demanding for big graphs)
157. Sampling = config['Sampling'] # Sampling parameters for big datasets
158. StatsOn=config['Parameters']['StatsOn']
159. Percolation_Basics = config['Percolation_Basics']
160.
161. # Variables
162. Number_Of_Nodes = config['N'] # Number of Nodes for each type of graph
163.
164. # ## Chapter 1 - Functions:
165. # This function takes a graph, plot name and target path as inputs,
166. # draws a degree histogram and a graph as a subplot (if DrawSubplot
167. # parameter is set to true) and saves it as an image to the location
168. # specified in config.txt as "OutFiguresPath" parameter
169. def DrawGraph (G, FigNumber, PlotName, Path, DrawSubgraph = "True"):
170.     fig, ax = plt.subplots()
171.     plt.style.use ('seaborn')
172.     degrees = [G.degree(n) for n in G.nodes()]
173.     degree_sequence = sorted([d for n, d in G.degree()], reverse=True)
174.     degreeCount = collections.Counter(degree_sequence)
175.     deg, cnt = zip(*degreeCount.items())
176.     # log scale for large networks
177.     if len(degreeCount)>50:

```

```

178.     print("Degree Count > 50 -> Log Scale")
179.     plt.bar(deg, cnt, color='grey', alpha=0.7)
180.     plt.yscale("log")
181.     plt.xscale("log")
182. else:
183.     plt.bar(deg, cnt, color='grey', alpha=0.7)
184.     locs, labels = plt.xticks()
185.     locations=[]
186.     for loc in locs:
187.         if (loc).is_integer():
188.             locations.append(int(loc))
189.         else:
190.             locations.append("")
191.     ax.set_xticklabels(locations)
192.     locs, labels = plt.yticks()
193.     locations=[]
194.     for loc in locs:
195.         if (loc).is_integer():
196.             locations.append(int(loc))
197.         else:
198.             locations.append("")
199.     ax.set_yticklabels(locations)
200. plt.title("Degree Histogram\n"+PlotName, fontsize=18)
201. plt.ylabel("Nodes", fontsize=16)
202. plt.xlabel("Degree", fontsize=16)
203. #fig.set_tight_layout(True)
204.
205. if DrawSubgraph == "True":
206.     # draw graph in inset
207.     plt.axes([0.25, 0.25, 0.5, 0.5])
208.     Gcc = G.subgraph(sorted(nx.connected_components(G), key=len,
reverse=True)[0])
209.     pos = nx.spring_layout(Gcc)
210.     plt.axis('off')
211.     nx.draw_networkx_nodes(Gcc, pos, node_size=20)
212.     nx.draw_networkx_edges(Gcc, pos, alpha=0.4)
213.     plt.savefig(Path+FigNumber+PlotName+".png", dpi=200, facecolor='w',
edgecolor='w')
214.     plt.clf()
215.
216. # -----Compute graph basic statistics -----
217. def ComputeStats(G, StatsOn=True):
218.     if StatsOn == True:
219.         print("Computing stats")
220.         NetworkDensity = nx.density(G)
221.         AvgPLength = nx.average_shortest_path_length(G, weight=None)
222.         NetworkDiameter = nx.diameter(G)
223.         AvgClustering = nx.average_clustering(G)
224.         print (nx.info(G))
225.         print ("Network Density: " + str(NetworkDensity))
226.         print ("Average Shortest Path Length: " + str(AvgPLength))
227.         print ("Network Diameter: " + str(NetworkDiameter))
228.         print ("Average Clustering Coefficient: " +
str(AvgClustering)+"\n")
229.     else:
230.         print ("Skipping stats. To enable set StatsOn to True in
[Parameters] section of config.txt")
231.
232. # -----working with samples of real online social networks -----
-----
233. # Construct graph from txt (big files, random sampling, skip first n
lines)
234. def EdgelistFromFile(filename, randomsample, BigFile, s, StartLine):

```

```

235.     # Returns list of tuples
236.     # Inputs:
237.     # Filename - full path
238.     InputNX = []
239.     G = nx.Graph()
240.     f = open(filename, 'r')
241.     for i in range(1,StartLine):
242.         f.readline()
243.     for i in range(1,s):
244.         Edge = f.readline()
245.         Edge = tuple(Edge.split())
246.         InputNX.append(Edge)
247.     print("Raw Graph ready")
248.     return InputNX
249.
250. # -----Trim small components from a graph G -----
251. def BiggestComponent(G):
252.     if len(list(nx.connected_components(G))) >1:
253.         BiggestComponent = sorted(list(nx.connected_components(G)),
key=len, reverse=True)[0]
254.         i = 0
255.         for component in list(nx.connected_components(G)):
256.             if len(component)<len(BiggestComponent):
257.                 for node in component:
258.                     G.remove_node(node)
259.                     i +=1
260.             print (str(i) + " nodes from smaller components trimmed")
261.         else:
262.             print ("Graph has one component (nothing to trim)")
263.
264. def ConstructGraph(type,s=0):
265.     randomsample = Sampling[type+"_random"]
266.     BigFile = Sampling[type+"_bigfile"]
267.
268.     if s==0:
269.         try:
270.             s = int(Sampling[type+"_sample"])
271.         except KeyError:
272.             print("Sample size set to 0. Add parameter to config.txt or
use an optional argument s")
273.     startline = int(Sampling[type+"_startline"])
274.     Edgelist = []
275.     Edgelist =
EdgelistFromFile(Path_In_Datasets+InFileNames[type],randomsample,
BigFile,s,startline)
276.     G=nx.Graph()
277.     G.add_edges_from(Edgelist)
278.     print("Total Nodes: "+ str(G.number_of_nodes()))
279.     BiggestComponent(G)
280.     print("Nodes left: "+ str(G.number_of_nodes()))
281.     N = G.number_of_nodes()
282.     Name = "Twitter Sample Graph (" +str(N)+" nodes)"
283.     s = 0
284.     return G,N
285. # ----- SIR and SIS models -----
286. def Percolation_Basics_Get_Inputs (type):
287.     global N
288.     try:
289.         N = int(Number_Of_Nodes[type])
290.     except KeyError:
291.         print(f"N for {type} not found in config.txt, proceeding with N
equal to {N}")
292.     Name = FigNames[type]

```

```

293.     rho_default = float(Percolation_Basics['rho_default'])
294.     p = float(Percolation_Basics['p'])
295.     iterations = int(Percolation_Basics['iterations'])
296.     tau = float(Percolation_Basics['tau'])
297.     gamma = float(Percolation_Basics['gamma'])
298.     tmax = int(Percolation_Basics['tmax'])
299.     try:
300.         rho = float(Percolation_Basics['rho'])
301.     except KeyError:
302.         rho = rho_default/N
303.     return N, Name, rho_default,p,iterations, tau,gamma,tmax,rho
304.
305. def Basic_Percolation(NW_Name,Type,NamePrefix="",NameSuffix=""):
306.     for counter in range(5):
307.         print("Basic "+type+" percolation. iteration "+str(counter+1))
308.         if Type == "SIR":
309.             t, S, I, R = EoN.fast_SIR(G, tau, gamma, rho=rho, tmax =
tmax)
310.             if counter == 1:
311.                 plt.plot(t, S, color = 'blue', alpha=0.3, label =
'Susceptible')
312.                 plt.plot(t, I, color = 'red', alpha=0.3, label =
'Infected')
313.                 plt.plot(t, R, color = 'black', alpha=0.3, label =
'Removed/Recovered')
314.                 plt.plot(t, S, color = 'blue', alpha=0.3)
315.                 plt.plot(t, I, color = 'red', alpha=0.3)
316.                 plt.plot(t, R, color = 'black', alpha=0.3)
317.             elif Type == "SIS":
318.                 t, S, I = EoN.fast_SIS(G, tau, gamma, rho=rho, tmax = tmax)
319.                 if counter == 1:
320.                     plt.plot(t, S, color = 'blue', alpha=0.3, label =
'Susceptible')
321.                     plt.plot(t, I, color = 'red', alpha=0.3, label =
'Infected')
322.                     plt.plot(t, S, color = 'blue', alpha=0.3)
323.                     plt.plot(t, I, color = 'red', alpha=0.3)
324.                 else:
325.                     print("Type can be only SIS/SIR")
326.             plt.xlabel('$t$')
327.             plt.ylabel('Number of Nodes')
328.             plt.title(str(iterations)+" "
+Type + ' simulations with '
+str(int(rho*N))
+' nodes initially infected on '
+str(NW_Name)+str(NameSuffix))
333.         plt.grid(True)
334.         plt.legend()
335.         plt.style.use ('seaborn')
336.         plt.savefig(Path_Out_Figs
+str(NamePrefix)
+Type+"_N"
+str(N)
+"_"+NW_Name
+'_rho_'
+str(rho)
+str(NameSuffix)
+'.png',
figsize=(15,7.5),
dpi= 200)
347.     plt.clf()
348.

```

```

349. def Percolation_Scalability_Test(NW_Name, PercolationType, GraphType,
350. min_s, max_s, step, NamePrefix="", NameSuffix="", Scaled=False):
351.     NetworkName = str(max_s//step)+" "+NW_Name+" Graphs"
352.     for N in range(min_s, max_s, step):
353.         if GraphType == "Barabasi":
354.             G = nx.barabasi_albert_graph(N, 2)
355.             N_nodes = N
356.         elif GraphType == "Custom_From_File":
357.             G, N = ConstructGraph(type, s=N)
358.             N_nodes = nx.number_of_nodes(G)
359.         if Scaled == True:
360.             D = N_nodes
361.             plt.ylabel('Fraction of Nodes')
362.         else:
363.             plt.ylabel('Number of Nodes')
364.             D = 1
365.         for counter in range(iterations):
366.             print("Basic "+type+" percolation. iteration
367. "+str(counter+1))
368.             if PercolationType == "SIR":
369.                 t, S, I, R = EoN.fast_SIR(G, tau, gamma, rho=rho, tmax =
370. tmax)
371.                 if counter == 1 and N==min_s:
372.                     plt.plot(t, S/D, color = 'blue', alpha=0.3, label =
373. 'Susceptible')
374.                     plt.plot(t, I/D, color = 'red', alpha=0.3, label =
375. 'Infected')
376.                     plt.plot(t, R/D, color = 'black', alpha=0.3, label =
377. 'Removed/Recovered')
378.                     plt.plot(t, S/D, color = 'blue', alpha=0.3)
379.                     plt.plot(t, I/D, color = 'red', alpha=0.3)
380.                     plt.plot(t, R/D, color = 'black', alpha=0.3)
381.                 elif PercolationType == "SIS":
382.                     t, S, I = EoN.fast_SIS(G, tau, gamma, rho=rho, tmax =
383. tmax)
384.                     if counter == 1 and N==min_s:
385.                         plt.plot(t, S/D, color = 'blue', alpha=0.3, label =
386. 'Susceptible')
387.                         plt.plot(t, I/D, color = 'red', alpha=0.3, label =
388. 'Infected')
389.                         plt.plot(t, S/D, color = 'blue', alpha=0.3)
390.                         plt.plot(t, I/D, color = 'red', alpha=0.3)
391.                     else:
392.                         print("Type can be only SIS/SIR")
393.                 G.clear()
394.                 if Scaled==True:
395.                     NameSuffix+=" (normalized)"
396.                 plt.xlabel('$t$')
397.                 plt.title(str(iterations)+" "
398. +PercolationType + ' simulations with '
399. +str(int(rho*N))
400. +' nodes initially infected on '
401. +str(NW_Name)+str(NameSuffix))
402.                 plt.grid(True)
403.                 plt.legend()
404.                 plt.style.use('seaborn')
405.                 plt.savefig(Path_Out_Figs
406. +str(NamePrefix)
407. +PercolationType+"_N"
408. +str(N_nodes)
409. +"_"+NW_Name
410. +'_rho_')

```

```

403.         +str(rho)
404.         +str(NameSuffix)
405.         +'.png',
406.         figsize=(15,7.5),
407.         dpi= 200)
408.     plt.clf()
409. # ## -----Execution -----
410. # Figure - Path graph:
411. type = "Path"
412. FigNumber = "Figure_1_"
413. G = nx.path_graph(int(Number_Of_Nodes[type]))
414. DrawGraph(G, FigNumber, FigNames[type]+" "+ Number_Of_Nodes[type]+"
Nodes", Path_Out_Figs, DrawSubgraph = DrawSubgraph)
415. #ComputeStats(G)
416. G.clear()
417. # Figure - Star graph:
418. type = "Star"
419. FigNumber = "Figure_2_"
420. G=nx.Graph()
421. for x in range(1,int(Number_Of_Nodes[type])):
422.     G.add_edge(0,x)
423. DrawGraph(G, FigNumber, FigNames[type]+" "+ Number_Of_Nodes[type]+"
Nodes", Path_Out_Figs, DrawSubgraph = DrawSubgraph)
424. #ComputeStats(G)
425. G.clear()
426. # Figure - Complete graph:
427. type = "Complete"
428. FigNumber = "Figure_3_"
429. G = nx.complete_graph(int(Number_Of_Nodes[type]))
430. DrawGraph(G, FigNumber, FigNames[type]+" "+ Number_Of_Nodes[type]+"
Nodes", Path_Out_Figs, DrawSubgraph = DrawSubgraph)
431. #ComputeStats(G)
432. G.clear()
433. # Figure - Facebook Graph - stats and degree distribution (Set
DrawGraphOnTopOfHist parameter to True in config.txt)
434. type = "FB"
435. FigNumber = "Figure_4_"
436. G = nx.read_edgelist(Path_In_Datasets+InFileNames[type], create_using =
nx.Graph(), nodetype = int)
437. N = nx.number_of_nodes(G)
438. DrawGraph(G, FigNumber, FigNames[type]+" "+ str(N)+" Nodes", Path_Out_Figs)
439. #ComputeStats(G)
440. G.clear()
441. # Figure - Twitter graph - sampling, stats and degree distribution
442. FigNumber = "Figure_5_"
443. type="TWTR"
444. G,N = ConstructGraph(type)
445. DrawGraph(G, FigNumber, FigNames[type]+" "+ str(N)+" Nodes", Path_Out_Figs)
446. #ComputeStats(G)
447. G.clear()
448. # Figure - LifeJournal graph - sampling, stats and degree distribution
449. FigNumber = "Figure_6_"
450. type="LJ" # Get Parameters from config.txt:
451. G,N = ConstructGraph(type)
452. DrawGraph(G, FigNumber, FigNames[type]+" "+ str(N)+" Nodes", Path_Out_Figs)
453. #ComputeStats(G)
454. G.clear()
455. # Figure - SIR on a random graph
456. type = "Random"
457. FigNumber = "Figure_7_"
458. N, Name, rho_default, p, iterations, tau, gamma, tmax, rho =
Percolation_Basics_Get_Inputs(type)
459. G = nx.fast_gnp_random_graph(N, p, seed=None, directed=False)

```

```

460. #DrawGraph(G,"",FigNumber,FigNames[type]+" "+str(N)+"
Nodes",Path_Out_Figs)
461. #ComputeStats(G)
462. Basic_Percolation(FigNames[type],Type = "SIR",NamePrefix=FigNumber)
463. G.clear()
464. # Figure - SIS on a random graph
465. type = "Random"
466. FigNumber = "Figure_8_"
467. N, Name, rho_default,p,iterations, tau,gamma,tmax,rho =
Percolation_Basics_Get_Inputs(type)
468. G = nx.fast_gnp_random_graph(N, p, seed=None, directed=False)
469. #DrawGraph(G,"",FigNumber,FigNames[type]+" "+str(N)+"
Nodes",Path_Out_Figs)
470. #ComputeStats(G)
471. Basic_Percolation(FigNames[type],Type = "SIS",NamePrefix=FigNumber)
472. G.clear()
473. # Figures SIR and SIS on Barabasi-Albert random graphs
474. type = "Barabasi"
475. N, Name, rho_default,p,iterations, tau,gamma,tmax,rho =
Percolation_Basics_Get_Inputs(type)
476. Barabasi_Edges = Percolation_Basics['barabasi_edges'].split(",")
477. Barabasi_Edges = [int(i) for i in Barabasi_Edges]
478. count=8
479. for i in Barabasi_Edges:
480.     count+=1
481.     NamePrefix = "Figure_"+str(count)+"_"
482.     NameSuffix = "_m_"+str(i)
483.     G = nx.barabasi_albert_graph(N,i)
484.     DrawGraph(G,NamePrefix,FigNames[type]+" "+str(N)+"
Nodes",Path_Out_Figs)
485.     #ComputeStats(G)
486.     count+=1
487.     NamePrefix = "Figure_"+str(count)+"_"
488.     Basic_Percolation(FigNames[type],Type =
"SIR",NamePrefix=NamePrefix,NameSuffix = NameSuffix)
489.     count+=1
490.     NamePrefix = "Figure_"+str(count)+"_"
491.     Basic_Percolation(FigNames[type],Type =
"SIS",NamePrefix=NamePrefix,NameSuffix = NameSuffix)
492.     G.clear()
493. # Figure 100 SIR simulations on 10 different Barabasi-Albert graphs
494. type = "Barabasi"
495. min_s = 1000
496. max_s = 11000
497. step = 1000
498. NamePrefix = "Figure_18_"
499. N, Name, rho_default,p,iterations, tau,gamma,tmax,rho =
Percolation_Basics_Get_Inputs(type)
500. Percolation_Scalability_Test("Barabasi-Albert Graphs with m =
2",PercolationType="SIR", GraphType=type, min_s=min_s, max_s=max_s,
501.     step=step,
NamePrefix=NamePrefix,NameSuffix="",Scaled=False)
502. # Figure 100 SIR simulations on 10 different Barabasi-Albert graphs
(normalized)
503. NamePrefix = "Figure_19_"
504. Percolation_Scalability_Test("Barabasi-Albert Graphs with m =
2",PercolationType="SIR", GraphType=type, min_s=min_s, max_s=max_s,
505.     step=step,
NamePrefix=NamePrefix,NameSuffix="",Scaled=True)
506. # Figure 100 SIS simulations on 10 different Barabasi-Albert graphs
507. NamePrefix = "Figure_20_"
508. Percolation_Scalability_Test("Barabasi-Albert Graphs with m =
2",PercolationType="SIS", GraphType=type, min_s=min_s, max_s=max_s,

```

```

509.                                     step=step,
    NamePrefix=NamePrefix,NameSuffix="",Scaled=False)
510. # Figure 100 SIS simulations on 10 different Barabasi-Albert graphs
    (normalized)
511. NamePrefix = "Figure_21_"
512. Percolation_Scalability_Test("Barabasi-Albert Graphs with m =
    2",PercolationType="SIS", GraphType=type, min_s=min_s, max_s=max_s,
513.                               step=step,
    NamePrefix=NamePrefix,NameSuffix="",Scaled=True)
514. # Figure 100 SIR simulations on 10 different Twitter graphs
515. type = "TWTR_Scalability"
516. min_s = 20000
517. max_s = 220000
518. step = 20000
519. NamePrefix = "Figure_22_"
520. Percolation_Scalability_Test("Twitter graphs",PercolationType="SIR",
    GraphType="Custom_From_File", min_s=min_s, max_s=max_s,
521.                               step=step,
    NamePrefix=NamePrefix,NameSuffix="",Scaled=False)
522. # Figure 100 SIR simulations on 10 Twitter graphs (normalized)
523. NamePrefix = "Figure_23_"
524. Percolation_Scalability_Test("Twitter graphs",PercolationType="SIR",
    GraphType="Custom_From_File", min_s=min_s, max_s=max_s,
525.                               step=step,
    NamePrefix=NamePrefix,NameSuffix="",Scaled=True)
526. # Figure 100 SIS simulations on 10 Twitter graphs
527. NamePrefix = "Figure_24_"
528. Percolation_Scalability_Test("Twitter graphs",PercolationType="SIS",
    GraphType="Custom_From_File", min_s=min_s, max_s=max_s,
529.                               step=step,
    NamePrefix=NamePrefix,NameSuffix="",Scaled=False)
530. # Figure 100 SIS simulations on 10 Twitter graphs (normalized)
531. NamePrefix = "Figure_25_"
532. Percolation_Scalability_Test("Twitter graphs",PercolationType="SIS",
    GraphType="Custom_From_File", min_s=min_s, max_s=max_s,
533.                               step=step,
    NamePrefix=NamePrefix,NameSuffix="",Scaled=True)
534. # Figure 100 SIR simulations on 10 LifeJournal graphs
535. type = "LJ_Scalability"
536. min_s = 1000
537. max_s = 11000
538. step = 1000
539. NamePrefix = "Figure_26_"
540. Percolation_Scalability_Test("LifeJournal graphs",PercolationType="SIR",
    GraphType="Custom_From_File", min_s=min_s, max_s=max_s,
541.                               step=step,
    NamePrefix=NamePrefix,NameSuffix="",Scaled=False)
542. # Figure 100 SIR simulations on 10 LifeJournal graphs (normalized)
543. NamePrefix = "Figure_27_"
544. Percolation_Scalability_Test("LifeJournal graphs",PercolationType="SIR",
    GraphType="Custom_From_File", min_s=min_s, max_s=max_s,
545.                               step=step,
    NamePrefix=NamePrefix,NameSuffix="",Scaled=True)
546. # Figure 100 SIS simulations on 10 LifeJournal graphs
547. NamePrefix = "Figure_28_"
548. Percolation_Scalability_Test("LifeJournal graphs",PercolationType="SIS",
    GraphType="Custom_From_File", min_s=min_s, max_s=max_s,
549.                               step=step,
    NamePrefix=NamePrefix,NameSuffix="",Scaled=False)
550. # Figure 100 SIS simulations on 10 LifeJournal graphs
551. NamePrefix = "Figure_29_"
552. Percolation_Scalability_Test("LifeJournal graphs",PercolationType="SIS",
    GraphType="Custom_From_File", min_s=min_s, max_s=max_s,

```

```

553.                                     step=step,
    NamePrefix=NamePrefix, NameSuffix="", Scaled=True)

```

```

1.  ## Tests on path graphs
2.  import networkx as nx
3.  import EoN
4.  import matplotlib.pyplot as plt
5.  import pandas as pd
6.  from collections import Counter
7.  tmax = 100
8.  iterations = 10000
9.  G = nx.path_graph(100)
10. initial_infections = [0]
11. #print(G.nodes)
12. simulations=[]
13. for counter in range(iterations):
14.     sim = EoN.fast_SIR(G, 0.7, 0, initial_infecteds = initial_infections,
15.         return_full_data=True, tmax = tmax)
16.     simulations.append(sim)
17. I = []
18. for sim in simulations:
19.     node_stats = sim.get_statuses(time=tmax)
20.     I.append ( Counter(node_stats.values())["I"])
21. df_I = pd.DataFrame(I)
22. #print(df_I)
23. #print(df_I.describe())
24. print(df_I.mean())
25. print(df_I.std())
26. G = nx.path_graph(100)
27. sim = EoN.fast_SIR(G, 0.5, 0, initial_infecteds = initial_infections,
28.     return_full_data=True, tmax = tmax)
29. sim.display(100)
30. import numpy as np
31. import math
32. data = I
33. bins = np.linspace(math.ceil(min(data)),
34.     math.floor(max(data)),
35.     100) # fixed number of bins
36. plt.xlim([min(data)-5, max(data)+5])
37. plt.hist(data, bins=bins, alpha=0.5)
38. #plt.title('Distribution of the quantity of infected nodes at t=100 (10
    000 simulations)')
39. plt.xlabel('nodes infected')
40. plt.ylabel('frequency')
41. plt.show()
42. #Normality tests
43. from scipy.stats import shapiro
44. from scipy.stats import normaltest
45. # normality test 1
46. sh_data, p = shapiro(data)
47. print('Statistics=%.3f, p=%.3f' % (sh_data, p))
48. # interpret
49. alpha = 0.05
50. if p > alpha:
51.     print('Sample looks Gaussian (fail to reject H0)')
52. else:
53.     print('Sample does not look Gaussian (reject H0)')
54. # normality test 2
55. normtest_data, p = normaltest(data)
56. print('Statistics=%.3f, p=%.3f' % (normtest_data, p))
57. # interpret
58. alpha = 0.05
59. if p > alpha:

```

```

60. print('Sample looks Gaussian (fail to reject H0)')
61. else:
62. print('Sample does not look Gaussian (reject H0)')
63. from scipy.stats import kurtosis
64. from scipy.stats import skew
65. print("Skewnesss", skew(data))
66. print("Kurtosis", kurtosis(data))
67. #p-p plot
68. import scipy.stats as stats
69. import pylab
70. stats.probplot(data, dist="norm", plot=pylab)
71. pylab.show()
72. # 500 simulations on a path graph
73. import networkx as nx
74. import EoN
75. import matplotlib.pyplot as plt
76. import pandas as pd
77. from collections import Counter
78.
79.
80. tmax = 500
81. iterations = 500
82.
83. G = nx.path_graph(100)
84. initial_infections = [0]
85.
86. simulations=[]
87.
88. for counter in range(iterations):
89.     t,S,I,R = EoN.fast_SIR(G, 0.7, 0, initial_infecteds =
        initial_infections,
90.         return_full_data=False, tmax = tmax)
91.
92.     if counter == 1:
93.         plt.plot(t, S, color = 'blue', alpha=0.3, label = 'Susceptible')
94.         plt.plot(t, I, color = 'red', alpha=0.3, label = 'Infected')
95.         plt.plot(t, R, color = 'black', alpha=0.3, label =
        'Removed/Recovered')
96.         plt.plot(t, S, color = 'blue', alpha=0.3)
97.         plt.plot(t, I, color = 'red', alpha=0.3)
98.         plt.plot(t, R, color = 'black', alpha=0.3)
99. plt.axhline(y = 70 ,color='g',alpha=0.5, label="Mean I at t=100")
100. plt.axvline(x = 100 ,color='black',alpha=0.4)
101. plt.legend()
102. plt.show()

```

Appendix B. Python code – Chapter 2

```
1. import numpy as np
2.
3. def get_I_at_time_t (given_value,a_list):
4.     absolute_difference_function = lambda list_value : abs(list_value -
given_value)
5.     closest_value = min(a_list, key=absolute_difference_function)
6.     element_index = np.where(t==closest_value)[0][0]
7.     return I[element_index]/N
8.
9. def mean_val(lst):
10.     return sum(lst) / len(lst)
11.     #To construct a graph the code uses functions from Appendix A
12.     type = "FB"
13.
14.     G = nx.read_edgelist(Path_In_Datasets+InFileNames[type], create_using =
nx.Graph(), nodetype = int)
15.     N = nx.number_of_nodes(G)
16.     Name = "FB network of "+str(N)+" nodes"
17.
18.     iterations_gradient_descent = 15
19.
20.     Total_Populaiton = 101 #mln. - population of DACH (OECD stat)
21.     Percent_Susceptible = 0.587 #- % of population at the age between 20
and 65
22.     #We assume all of them make up the FB network.
23.
24.     iterations_sir = 100
25.     tau_init = 0.5
26.     tau_step = 0.25
27.     gamma = 0
28.     tmax = 30
29.     rho = 0.927482/(Total_Populaiton*Percent_Susceptible) #number of users
at the end of 2005 * total susceptible (assumed to be 90% of pop)
30.     Scaled = True
31.
32.     target_t = 14
33.     target_I = 17.24/(Total_Populaiton*Percent_Susceptible)
34.
35.     tau = tau_init
36.     counter_gd = 0
37.     while counter_gd < iterations_gradient_descent:
38.         infected = []
39.         print("Gradient Descent Step: ",counter_gd+1,"transmission
rate:",tau)
40.         for counter in range(iterations_sir):
41.             t, S, I, R = EoN.fast_SIR(G, tau, gamma,rho=rho, tmax = tmax)
42.             infected.append(get_I_at_time_t(target_t,t))
43.             if mean_val(Infected)<target_I:
44.                 tau = tau + tau_step
45.             else:
46.                 tau = tau - tau_step
47.                 tau_step = tau_step/2
48.                 counter_gd +=1
49.
50.     #final SIR simulations with target tau:
51.     D = 1
52.     if Scaled == True:
53.         D=N
54.
```

```

55.
56.     iterations_sir = 100
57.
58.     infected = []
59.     for counter in range(iterations_sir):
60.         t, S, I, R = EoN.fast_SIR(G, tau, gamma, rho=rho, tmax = tmax)
61.         if counter == 1:
62.             #plt.plot(t, S/D, color = 'blue', alpha=0.3, label =
'Susceptible')
63.             plt.plot(t, I/D, color = 'red', alpha=0.2, label = 'Infected')
64.             #plt.plot(t, R/D, color = 'black', alpha=0.3, label =
'Removed/Recovered')
65.             #plt.plot(t, S/D, color = 'blue', alpha=0.3)
66.             plt.plot(t, I/D, color = 'red', alpha=0.3)
67.             #plt.plot(t, R/D, color = 'black', alpha=0.3)
68.
69.             for c in range(29):
70.                 infected.append([c, get_I_at_time_t(c,t)])
71.
72.
73.     act_I = []
74.     for i in actual_I:
75.         act_I.append(i/(Total_Populaiton*Percent_Susceptible))
76.
77.     plt.plot(act_I, color = 'green', alpha = 0.8, label='Actual number of
members')
78.     plt.axvline(x = target_t, alpha=0.3)
79.     if Scaled == False:
80.         target_I = target_I * N
81.     plt.axhline(y = target_I, alpha=0.3)
82.     plt.legend()
83.     plt.xlabel('$t$')
84.     plt.ylabel('Number of Nodes')
85.     plt.title(str(iterations_sir)+" simulaitons of SIR process on "+Name)
86.     plt.show()
87.     #saving results to Excel
88.     XING_forecast_xlsx =
"C:/Users/Me/Desktop/New_folder/XING_Rev_forecast.xlsx"
89.     df = pd.DataFrame.from_records(infected, index=0)
90.     df.to_excel(XING_forecast_xlsx)

```

Appendix C. VBA code for generating triangular distributions in MS Excel

```
1. `Triangular distribution
2. Function TRIDIST(random As Double, min As Double, max As Double, mode
   As Double)
3.     If mode < min Or max < mode Then
4.         TRIDIST = CVErr(xlErrValue)
5.     Else
6.         If random <= (mode - min) / (max - min) Then
7.             TRIDIST = min + Sqr((max - min) * (mode - min) * random)
8.         Else
9.             TRIDIST = max - Sqr((max - min) * (max - mode) * (1 -
   random))
10.        End If
11.    End If
12. End Function
```

Appendix D. Cosine similarity matrixes by companies

AMZN	1997	1998	1999	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017	2018	2019
1997	1,00	0,61	0,33	0,33	0,31	0,36	0,31	0,30	0,29	0,30	0,31	0,30	0,30	0,30	0,29	0,27	0,28	0,28	0,28	0,28	0,27	0,30	0,28
1998	0,61	1,00	0,54	0,51	0,46	0,50	0,44	0,43	0,41	0,42	0,44	0,42	0,41	0,40	0,37	0,38	0,38	0,39	0,39	0,39	0,37	0,40	0,38
1999	0,33	0,54	1,00	0,87	0,79	0,77	0,75	0,71	0,69	0,70	0,69	0,69	0,68	0,65	0,65	0,59	0,61	0,60	0,61	0,60	0,56	0,58	0,57
2000	0,33	0,51	0,87	1,00	0,91	0,87	0,79	0,76	0,75	0,75	0,75	0,73	0,72	0,69	0,68	0,63	0,65	0,64	0,64	0,64	0,60	0,62	0,61
2001	0,31	0,46	0,79	0,91	1,00	0,92	0,85	0,79	0,77	0,76	0,75	0,73	0,72	0,70	0,70	0,65	0,66	0,66	0,66	0,66	0,61	0,63	0,62
2002	0,36	0,50	0,77	0,87	0,92	1,00	0,88	0,84	0,81	0,80	0,81	0,77	0,76	0,73	0,72	0,67	0,68	0,68	0,69	0,68	0,64	0,67	0,65
2003	0,31	0,44	0,75	0,79	0,85	0,88	1,00	0,92	0,88	0,87	0,84	0,84	0,83	0,81	0,80	0,74	0,75	0,75	0,75	0,74	0,69	0,70	0,70
2004	0,30	0,43	0,71	0,76	0,79	0,84	0,92	1,00	0,95	0,91	0,87	0,85	0,84	0,82	0,81	0,75	0,76	0,76	0,76	0,76	0,70	0,70	0,71
2005	0,29	0,41	0,69	0,75	0,77	0,81	0,88	0,95	1,00	0,96	0,90	0,87	0,85	0,83	0,82	0,76	0,76	0,76	0,77	0,76	0,71	0,71	0,71
2006	0,30	0,42	0,70	0,75	0,76	0,80	0,87	0,91	0,96	1,00	0,95	0,91	0,88	0,85	0,83	0,77	0,78	0,77	0,78	0,78	0,72	0,72	0,73
2007	0,31	0,44	0,69	0,75	0,75	0,81	0,84	0,87	0,90	0,95	1,00	0,95	0,91	0,85	0,83	0,77	0,78	0,78	0,78	0,78	0,73	0,74	0,73
2008	0,30	0,42	0,69	0,73	0,73	0,77	0,84	0,85	0,87	0,91	0,95	1,00	0,95	0,89	0,86	0,79	0,79	0,79	0,80	0,79	0,74	0,74	0,75
2009	0,30	0,42	0,68	0,72	0,72	0,76	0,83	0,84	0,85	0,88	0,91	0,95	1,00	0,95	0,91	0,82	0,82	0,82	0,82	0,81	0,76	0,76	0,77
2010	0,30	0,41	0,65	0,69	0,70	0,73	0,81	0,82	0,83	0,85	0,85	0,89	0,95	1,00	0,96	0,86	0,83	0,82	0,82	0,81	0,75	0,75	0,76
2011	0,29	0,40	0,65	0,68	0,70	0,72	0,80	0,81	0,82	0,83	0,83	0,86	0,91	0,96	1,00	0,92	0,87	0,84	0,83	0,82	0,76	0,76	0,77
2012	0,27	0,37	0,59	0,63	0,65	0,67	0,74	0,75	0,76	0,77	0,77	0,79	0,82	0,86	0,92	1,00	0,93	0,87	0,81	0,80	0,74	0,73	0,74
2013	0,28	0,38	0,61	0,65	0,66	0,68	0,75	0,76	0,76	0,78	0,78	0,79	0,82	0,83	0,87	0,93	1,00	0,94	0,87	0,81	0,75	0,74	0,75
2014	0,28	0,38	0,60	0,64	0,66	0,68	0,75	0,76	0,76	0,77	0,78	0,79	0,82	0,82	0,84	0,87	0,94	1,00	0,94	0,87	0,77	0,76	0,77
2015	0,28	0,39	0,61	0,64	0,66	0,69	0,75	0,76	0,77	0,78	0,78	0,80	0,82	0,82	0,83	0,81	0,87	0,94	1,00	0,94	0,82	0,77	0,78
2016	0,28	0,39	0,60	0,64	0,66	0,68	0,74	0,76	0,76	0,78	0,78	0,79	0,81	0,81	0,82	0,80	0,81	0,87	0,94	1,00	0,88	0,82	0,79
2017	0,27	0,37	0,56	0,60	0,61	0,64	0,69	0,70	0,71	0,72	0,73	0,74	0,76	0,75	0,76	0,74	0,75	0,77	0,82	0,88	1,00	0,90	0,82
2018	0,30	0,40	0,58	0,62	0,63	0,67	0,70	0,70	0,71	0,72	0,74	0,74	0,76	0,75	0,76	0,73	0,74	0,76	0,77	0,82	0,90	1,00	0,91
2019	0,28	0,38	0,57	0,61	0,62	0,65	0,70	0,71	0,71	0,73	0,73	0,75	0,77	0,76	0,77	0,74	0,75	0,77	0,78	0,79	0,82	0,91	1,00

ATVI	1995	1996	1997	1998	1999	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017	2018	2019
1995	1,00	0,87	0,77	0,72	0,40	0,66	0,43	0,49	0,57	0,58	0,57	0,57	0,55	0,49	0,52	0,51	0,52	0,53	0,52	0,50	0,50	0,45	0,43	0,42	0,44
1996	0,87	1,00	0,90	0,84	0,44	0,73	0,46	0,52	0,58	0,59	0,58	0,58	0,56	0,52	0,54	0,54	0,56	0,56	0,56	0,54	0,54	0,49	0,47	0,46	0,48
1997	0,77	0,90	1,00	0,94	0,47	0,76	0,48	0,54	0,60	0,61	0,60	0,60	0,58	0,55	0,57	0,56	0,58	0,59	0,58	0,56	0,56	0,52	0,49	0,49	0,51
1998	0,72	0,84	0,94	1,00	0,49	0,80	0,50	0,56	0,63	0,63	0,62	0,62	0,60	0,58	0,59	0,59	0,60	0,61	0,60	0,58	0,58	0,53	0,51	0,50	0,53
1999	0,40	0,44	0,47	0,49	1,00	0,71	0,91	0,68	0,38	0,37	0,36	0,37	0,36	0,36	0,36	0,36	0,36	0,36	0,38	0,37	0,38	0,33	0,32	0,31	0,33
2000	0,66	0,73	0,76	0,80	0,71	1,00	0,69	0,82	0,61	0,59	0,58	0,58	0,57	0,55	0,57	0,56	0,57	0,57	0,57	0,56	0,55	0,50	0,47	0,46	0,48
2001	0,43	0,46	0,48	0,50	0,91	0,69	1,00	0,62	0,45	0,42	0,42	0,42	0,41	0,40	0,40	0,40	0,40	0,40	0,41	0,40	0,41	0,37	0,35	0,34	0,36
2002	0,49	0,52	0,54	0,56	0,68	0,82	0,62	1,00	0,60	0,55	0,53	0,54	0,52	0,50	0,51	0,50	0,50	0,49	0,50	0,48	0,49	0,43	0,41	0,40	0,42
2003	0,57	0,58	0,60	0,63	0,38	0,61	0,45	0,60	1,00	0,88	0,84	0,83	0,80	0,68	0,70	0,69	0,68	0,68	0,66	0,63	0,65	0,60	0,57	0,57	0,60
2004	0,58	0,59	0,61	0,63	0,37	0,59	0,42	0,55	0,88	1,00	0,95	0,89	0,83	0,68	0,69	0,69	0,69	0,69	0,66	0,64	0,65	0,60	0,57	0,57	0,60
2005	0,57	0,58	0,60	0,62	0,36	0,58	0,42	0,53	0,84	0,95	1,00	0,95	0,88	0,70	0,70	0,69	0,69	0,69	0,66	0,63	0,65	0,60	0,57	0,57	0,60
2006	0,57	0,58	0,60	0,62	0,37	0,58	0,42	0,54	0,83	0,89	0,95	1,00	0,94	0,74	0,73	0,70	0,70	0,67	0,64	0,65	0,61	0,58	0,58	0,60	0,60
2007	0,55	0,56	0,58	0,60	0,36	0,57	0,41	0,52	0,80	0,83	0,88	0,94	1,00	0,83	0,81	0,76	0,73	0,73	0,70	0,66	0,68	0,62	0,59	0,58	0,61
2008	0,49	0,52	0,55	0,58	0,36	0,55	0,40	0,50	0,68	0,68	0,70	0,74	0,83	1,00	0,96	0,90	0,83	0,83	0,79	0,76	0,76	0,70	0,67	0,67	0,71
2009	0,52	0,54	0,57	0,59	0,36	0,57	0,40	0,51	0,70	0,69	0,70	0,73	0,81	0,96	1,00	0,95	0,88	0,85	0,81	0,77	0,78	0,72	0,69	0,68	0,72
2010	0,51	0,54	0,56	0,59	0,36	0,56	0,40	0,50	0,69	0,69	0,69	0,70	0,76	0,90	0,95	1,00	0,95	0,90	0,82	0,78	0,78	0,72	0,69	0,69	0,73
2011	0,52	0,56	0,58	0,60	0,36	0,57	0,40	0,50	0,68	0,69	0,69	0,70	0,73	0,83	0,88	0,95	1,00	0,96	0,86	0,79	0,79	0,73	0,70	0,70	0,73
2012	0,53	0,56	0,59	0,61	0,36	0,57	0,40	0,49	0,68	0,69	0,69	0,70	0,73	0,83	0,85	0,90	0,96	1,00	0,92	0,83	0,80	0,73	0,71	0,70	0,74
2013	0,52	0,56	0,58	0,60	0,38	0,57	0,41	0,50	0,66	0,66	0,66	0,67	0,70	0,79	0,81	0,82	0,86	0,92	1,00	0,94	0,85	0,74	0,71	0,70	0,73
2014	0,50	0,54	0,56	0,58	0,37	0,56	0,40	0,48	0,63	0,64	0,63	0,64	0,66	0,76	0,77	0,78	0,79	0,83	0,94	1,00	0,92	0,79	0,71	0,70	0,73
2015	0,50	0,54	0,56	0,58	0,38	0,55	0,41	0,49	0,65	0,65	0,65	0,65	0,68	0,76	0,78	0,78	0,79	0,80	0,85	0,92	1,00	0,91	0,80	0,74	0,76
2016	0,45	0,49	0,52	0,53	0,33	0,50	0,37	0,43	0,60	0,60	0,60	0,61	0,62	0,70	0,72	0,72	0,73	0,73	0,74	0,79	0,91	1,00	0,91	0,81	0,75
2017	0,43	0,47	0,49	0,51	0,32	0,47	0,35	0,41	0,57	0,57	0,57	0,58	0,59	0,67	0,69	0,69	0,70	0,71	0,71	0,71	0,80	0,91	1,00	0,93	0,79
2018	0,42	0,46	0,49	0,50	0,31	0,46	0,34	0,40	0,57	0,57	0,57	0,58	0,58	0,67	0,68	0,69	0,70	0,70	0,70	0,74	0,81	0,93	1,00	0,88	0,78
2019	0,44	0,48	0,51	0,53	0,33	0,48	0,36	0,42	0,60	0,60	0,60	0,60	0,61	0,71	0,72	0,73	0,73	0,74	0,73	0,73	0,76	0,75	0,79	0,88	1,00

EBAY	1998	1999	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017	2018	2019		
1998	1,00	0,62	0,71	0,71	0,64	0,64	0,65	0,59	0,57	0,58	0,58	0,57	0,55	0,54	0,53	0,53	0,52	0,53	0,52	0,51	0,53	0,51	0,53	0,51
1999	0,62	1,00	0,59	0,62	0,53	0,50	0,50	0,47	0,45	0,46	0,47	0,47	0,46	0,45	0,44	0,45	0,44	0,44	0,43	0,42	0,50	0,45	0,45	0

GOOGL	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017	2018	2019
2004	1,00	0,94	0,89	0,86	0,83	0,82	0,78	0,78	0,77	0,73	0,69	0,67	0,68	0,67	0,66	0,65
2005	0,94	1,00	0,95	0,92	0,88	0,87	0,81	0,81	0,79	0,75	0,71	0,69	0,70	0,69	0,69	0,67
2006	0,89	0,95	1,00	0,96	0,91	0,89	0,83	0,82	0,81	0,76	0,72	0,70	0,71	0,71	0,70	0,69
2007	0,86	0,92	0,96	1,00	0,95	0,91	0,83	0,83	0,81	0,77	0,72	0,70	0,71	0,71	0,70	0,69
2008	0,83	0,88	0,91	0,95	1,00	0,95	0,87	0,84	0,81	0,77	0,73	0,71	0,72	0,72	0,72	0,70
2009	0,82	0,87	0,89	0,91	0,95	1,00	0,91	0,87	0,82	0,78	0,73	0,71	0,72	0,73	0,72	0,70
2010	0,78	0,81	0,83	0,83	0,87	0,91	1,00	0,96	0,89	0,84	0,81	0,77	0,78	0,78	0,77	0,75
2011	0,78	0,81	0,82	0,83	0,84	0,87	0,96	1,00	0,94	0,88	0,83	0,78	0,79	0,79	0,78	0,76
2012	0,77	0,79	0,81	0,81	0,81	0,82	0,89	0,94	1,00	0,95	0,88	0,80	0,80	0,80	0,79	0,77
2013	0,73	0,75	0,76	0,77	0,77	0,78	0,84	0,88	0,95	1,00	0,92	0,82	0,80	0,79	0,78	0,76
2014	0,69	0,71	0,72	0,72	0,73	0,73	0,81	0,83	0,88	0,92	1,00	0,87	0,83	0,79	0,77	0,75
2015	0,67	0,69	0,70	0,70	0,71	0,71	0,77	0,78	0,80	0,82	0,87	1,00	0,94	0,88	0,83	0,80
2016	0,68	0,70	0,71	0,71	0,72	0,72	0,78	0,79	0,80	0,80	0,83	0,94	1,00	0,96	0,88	0,83
2017	0,67	0,69	0,71	0,71	0,72	0,73	0,78	0,79	0,80	0,79	0,79	0,88	0,96	1,00	0,94	0,86
2018	0,66	0,69	0,70	0,70	0,72	0,72	0,77	0,78	0,79	0,78	0,77	0,83	0,88	0,94	1,00	0,93
2019	0,65	0,67	0,69	0,69	0,70	0,70	0,75	0,76	0,77	0,76	0,75	0,80	0,83	0,86	0,93	1,00

MSFT	1994	1995	1996	1997	1998	1999	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017	2018	2019	
1994	1,00	0,63	0,47	0,73	0,41	0,71	0,84	0,60	0,35	0,31	0,33	0,34	0,32	0,32	0,32	0,32	0,31	0,32	0,31	0,29	0,29	0,29	0,29	0,28	0,28	0,28	
1995	0,63	1,00	0,90	0,82	0,81	0,72	0,62	0,77	0,67	0,54	0,56	0,56	0,53	0,54	0,54	0,53	0,53	0,54	0,54	0,51	0,50	0,50	0,47	0,47	0,47	0,47	
1996	0,47	0,90	1,00	0,82	0,90	0,70	0,51	0,76	0,71	0,57	0,58	0,58	0,55	0,56	0,56	0,55	0,55	0,56	0,56	0,53	0,52	0,52	0,49	0,49	0,49	0,49	
1997	0,73	0,82	0,82	1,00	0,83	0,82	0,68	0,81	0,65	0,53	0,56	0,55	0,52	0,52	0,52	0,51	0,51	0,52	0,51	0,49	0,48	0,48	0,46	0,46	0,45	0,45	
1998	0,41	0,81	0,90	0,83	1,00	0,75	0,50	0,79	0,75	0,60	0,62	0,61	0,58	0,58	0,58	0,57	0,57	0,58	0,58	0,55	0,54	0,54	0,51	0,51	0,50	0,51	
1999	0,71	0,72	0,70	0,82	0,75	1,00	0,64	0,84	0,63	0,53	0,54	0,53	0,51	0,51	0,51	0,50	0,50	0,51	0,50	0,48	0,47	0,48	0,46	0,46	0,44	0,45	
2000	0,84	0,62	0,51	0,68	0,50	0,64	1,00	0,64	0,45	0,36	0,38	0,38	0,36	0,36	0,36	0,36	0,35	0,36	0,35	0,33	0,33	0,33	0,32	0,32	0,32	0,32	
2001	0,60	0,77	0,76	0,81	0,79	0,84	0,64	1,00	0,85	0,69	0,67	0,66	0,62	0,62	0,62	0,60	0,60	0,61	0,60	0,58	0,58	0,58	0,56	0,55	0,54	0,54	
2002	0,35	0,67	0,71	0,65	0,75	0,63	0,45	0,85	1,00	0,84	0,77	0,72	0,69	0,69	0,69	0,67	0,67	0,68	0,67	0,65	0,65	0,65	0,63	0,62	0,61	0,61	
2003	0,31	0,54	0,57	0,53	0,60	0,53	0,36	0,69	0,84	1,00	0,90	0,82	0,76	0,74	0,74	0,72	0,71	0,71	0,70	0,69	0,68	0,69	0,68	0,67	0,65	0,65	
2004	0,33	0,56	0,58	0,56	0,62	0,54	0,38	0,67	0,77	0,90	1,00	0,92	0,84	0,78	0,76	0,75	0,73	0,73	0,72	0,70	0,70	0,70	0,69	0,67	0,65	0,65	
2005	0,34	0,56	0,58	0,55	0,61	0,53	0,38	0,66	0,72	0,82	0,92	1,00	0,93	0,85	0,80	0,78	0,76	0,75	0,74	0,72	0,71	0,72	0,71	0,69	0,68	0,68	
2006	0,32	0,53	0,55	0,52	0,58	0,51	0,36	0,62	0,69	0,76	0,84	0,93	1,00	0,93	0,84	0,78	0,76	0,75	0,74	0,72	0,70	0,71	0,70	0,68	0,67	0,67	
2007	0,32	0,54	0,56	0,52	0,58	0,51	0,36	0,62	0,69	0,74	0,78	0,85	0,93	1,00	0,92	0,83	0,77	0,76	0,75	0,72	0,71	0,72	0,71	0,69	0,67	0,67	
2008	0,32	0,54	0,56	0,52	0,58	0,51	0,36	0,62	0,69	0,74	0,76	0,80	0,84	0,92	1,00	0,92	0,82	0,77	0,75	0,73	0,72	0,72	0,71	0,69	0,68	0,68	
2009	0,32	0,53	0,55	0,51	0,57	0,50	0,36	0,60	0,67	0,72	0,75	0,78	0,78	0,83	0,92	1,00	0,91	0,82	0,76	0,74	0,73	0,74	0,73	0,71	0,69	0,69	
2010	0,31	0,53	0,55	0,51	0,57	0,50	0,35	0,60	0,67	0,71	0,73	0,76	0,76	0,77	0,82	0,91	1,00	0,92	0,83	0,77	0,75	0,75	0,75	0,73	0,71	0,71	
2011	0,32	0,54	0,56	0,52	0,58	0,51	0,36	0,61	0,68	0,71	0,73	0,75	0,75	0,76	0,77	0,82	0,92	1,00	0,92	0,83	0,77	0,77	0,76	0,74	0,72	0,72	
2012	0,31	0,54	0,56	0,51	0,58	0,50	0,35	0,60	0,67	0,70	0,72	0,74	0,74	0,75	0,75	0,76	0,83	0,92	1,00	0,92	0,81	0,78	0,76	0,74	0,72	0,72	
2013	0,29	0,51	0,53	0,49	0,55	0,48	0,33	0,58	0,65	0,69	0,70	0,72	0,72	0,72	0,73	0,74	0,77	0,83	0,92	1,00	0,90	0,82	0,76	0,75	0,73	0,73	
2014	0,29	0,50	0,52	0,48	0,54	0,47	0,33	0,58	0,65	0,68	0,70	0,71	0,70	0,71	0,72	0,73	0,75	0,77	0,81	0,90	1,00	0,92	0,82	0,77	0,75	0,75	
2015	0,29	0,50	0,52	0,48	0,54	0,48	0,33	0,58	0,65	0,69	0,70	0,72	0,71	0,72	0,72	0,74	0,75	0,77	0,78	0,82	0,92	1,00	0,91	0,82	0,76	0,76	
2016	0,29	0,47	0,49	0,46	0,51	0,46	0,33	0,56	0,63	0,68	0,69	0,71	0,70	0,71	0,71	0,73	0,75	0,76	0,76	0,76	0,76	0,82	0,91	1,00	0,92	0,82	0,78
2017	0,28	0,47	0,49	0,46	0,51	0,46	0,32	0,55	0,62	0,67	0,67	0,69	0,68	0,69	0,69	0,71	0,73	0,74	0,74	0,75	0,77	0,82	0,92	1,00	0,91	0,84	
2018	0,28	0,47	0,49	0,45	0,50	0,44	0,32	0,54	0,61	0,65	0,65	0,68	0,67	0,67	0,68	0,69	0,71	0,72	0,72	0,73	0,75	0,76	0,82	0,91	1,00	0,94	
2019	0,28	0,47	0,49	0,45	0,51	0,45	0,32	0,54	0,61	0,65	0,65	0,68	0,67	0,67	0,68	0,69	0,71	0,72	0,72	0,73	0,75	0,76	0,82	0,91	1,00	0,94	

RDSA	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017	2018	2019
2005	1,00	0,97	0,86	0,83	0,70	0,78	0,78	0,77	0,75	0,70	0,76	0,71	0,70	0,69	0,71
2006	0,97	1,00	0,91	0,87	0,72	0,80	0,80	0,79	0,77	0,71	0,78	0,74	0,72	0,72	0,74
2007	0,86	0,91	1,00	0,92	0,74	0,80	0,80	0,80	0,78	0,71	0,78	0,75	0,74	0,73	0,75
2008	0,83	0,87	0,92	1,00	0,80	0,87	0,85	0,85	0,82	0,75	0,83	0,79	0,78	0,78	0,79
2009	0,70	0,72	0,74	0,80	1,00	0,83	0,78	0,76	0,73	0,66	0,73	0,69	0,68	0,67	0,68
2010	0,78	0,80	0,80	0,87	0,83	1,00	0,94	0,89	0,86	0,77	0,84	0,79	0,78	0,77	0,78
2011	0,78	0,80	0,80	0,85	0,78	0,94	1,00	0,94	0,88	0,78	0,84	0,80	0,79	0,78	0,78
2012	0,77	0,79	0,80	0,85	0,76	0,89	0,94	1,00	0,94	0,80	0,86	0,81	0,80	0,79	0,80
2013	0,75	0,77	0,78	0,82	0,73	0,86	0,88	0,94	1,00	0,86	0,88	0,81	0,80	0,79	0,80
2014	0,70	0,71	0,71	0,75	0,66	0,77	0,78	0,80	0,86	1,00	0,84	0,76	0,74	0,72	0,73
2015	0,76	0,78	0,78	0,83	0,73	0,84	0,84	0,86	0,88	0,84	1,00	0,89	0,85	0,82	0,83
2016	0,71	0,74	0,75	0,79	0,69	0,79	0,80	0,81	0,81	0,76	0,89	1,00	0,94	0,88	0,88
2017	0,70	0,72	0,74	0,78	0,68	0,78	0,79	0,80	0,80	0,74	0,85	0,94	1,00	0,94	0,89
2018	0,69	0,72	0,73	0,78	0,67	0,77	0,78	0,79	0,79	0,72	0,82	0,88	0,94	1,00	0,92
2019	0,71	0,74	0,75	0,79	0,68	0,78	0,78	0,80	0,80	0,73	0,83	0,88	0,89	0,92	1,00

XING	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017	2018	2019
2006	1,00	0,33	0,28	0,23	0,69	0,66	0,67	0,65	0,63	0,63	0,61	0,58	0,58	0,55
2007	0,33	1,00	0,48	0,37	0,25	0,25	0,26	0,25	0,25	0,25	0,24	0,22	0,21	0,18
2008	0,28	0,48	1,00	0,48	0,29	0,27	0,29	0,28	0,28	0,28	0,27	0,26	0,25	0,23
2009	0,23	0,37	0,48	1,00	0,30	0,27	0,27	0,26	0,26	0,26	0,25	0,23	0,22	0,19
2010	0,69	0,25	0,29	0,30	1,00	0,90	0,83	0,77	0,75	0,76	0,74	0,68	0,68	0,62
2011	0,66	0,25	0,27	0,27	0,90	1,00	0,87	0,76	0,73	0,74	0,71	0,64	0,64	0,58
2012	0,67	0,26	0,29	0,27	0,83	0,87	1,00	0,87	0,80	0,79	0,77	0,69	0,69	0,62
2013	0,65	0,25	0,28	0,26	0,77	0,76	0,87	1,00	0,88	0,83	0,81	0,73	0,72	0,66
2014	0,63	0,25	0,28	0,26	0,75	0,73	0,80	0,88	1,00	0,90	0,83	0,76	0,74	0,67
2015	0,63	0,25	0,28	0,26	0,76	0,74	0,79	0,83	0,90	1,00	0,89	0,77	0,75	0,68
2016	0,61	0,24	0,27	0,25	0,74	0,71	0,77	0,81	0,83	0,89	1,00	0,83	0,76	0,68
2017	0,58	0,22	0,26	0,23	0,68	0,64	0,69	0,73	0,76	0,77	0,83	1,00	0,86	0,72
2018	0,58	0,21	0,25	0,22	0,68	0,64	0,69	0,72	0,74	0,75	0,76	0,86	1,00	0,82
2019	0,55	0,18	0,23	0,19	0,62	0,58	0,62	0,66	0,67	0,68	0,68	0,72	0,82	1,00

XOM	1993	1994	1995	1996	1997	1998	1999	2000	2001	2002	2003	2004	2005	2006	2007	2008	2009	2010	2011	2012	2013	2014	2015	2016	2017	2018	2019
1993	1,00	0,84	0,86	0,69	0,80	0,74	0,71	0,66	0,66	0,66	0,64	0,64	0,62	0,63	0,61	0,61	0,60	0,58	0,60	0,60	0,58	0,57	0,57	0,56	0,55	0,58	
1994	0,84	1,00	0,90	0,64	0,63	0,83	0,55	0,78	0,77	0,77	0,75	0,75	0,72	0,73	0,71	0,71	0,67	0,67	0,69	0,69	0,66	0,65	0,65	0,66	0,64	0,63	0,67
1995	0,86	0,90	1,00	0,70	0,64	0,81	0,64	0,73	0,73	0,72	0,70	0,70	0,68	0,68	0,67	0,67	0,63	0,63	0,66	0,65	0,62	0,61	0,61	0,62	0,60	0,60	0,63
1996	0,69	0,64	0,70	1,00	0,54	0,64	0,91	0,55	0,54	0,52	0,51	0,50	0,49	0,49	0,48	0,48	0,47	0,47	0,46	0,48	0,46	0,45	0,43	0,45	0,42	0,42	0,44
1997	0,60	0,63	0,64	0,54	1,00	0,69	0,52	0,56	0,56	0,55	0,54	0,54	0,52	0,52	0,51	0,51	0,51	0,49	0,50	0,50	0,48	0,47	0,47	0,48	0,46	0,46	0,48
1998	0,74	0,83	0,81	0,64	0,69	1,00	0,60	0,83	0,79	0,77	0,75	0,74	0,72	0,72	0,71	0,71	0,66	0,67	0,69	0,69	0,66	0,65	0,65	0,66	0,64	0,63	0,66
1999	0,71	0,55	0,64	0,91	0,52	0,60	1,00	0,58	0,53	0,50	0,48	0,47	0,46	0,46	0,45	0,46	0,46	0,46	0,45	0,46	0,44	0,44	0,43	0,43	0,42	0,41	0,42
2000	0,66	0,78	0,73	0,55	0,56	0,83	0,58	1,00	0,94	0,88	0,83	0,82	0,79	0,79	0,77	0,78	0,72	0,74	0,76	0,76	0,73	0,72	0,72	0,71	0,70	0,72	0,72
2001	0,66	0,77	0,73	0,54	0,56	0,79	0,53	0,94	1,00	0,94	0,88	0,85	0,82	0,82	0,80	0,80	0,73	0,75	0,78	0,78	0,75	0,74	0,74	0,74	0,73	0,72	0,74
2002	0,66	0,77	0,72	0,52	0,55	0,77	0,50	0,88	0,94	1,00	0,94	0,89	0,84	0,84	0,82	0,82	0,76	0,77	0,80	0,80	0,77	0,76	0,76	0,76	0,74	0,74	0,77
2003	0,64	0,75	0,70	0,51	0,54	0,75	0,48	0,83	0,88	0,94	1,00	0,95	0,88	0,86	0,84	0,84	0,77	0,79	0,81	0,81	0,78	0,76	0,76	0,77	0,75	0,75	0,77
2004	0,64	0,75	0,70	0,50	0,54	0,74	0,47	0,82	0,85	0,89	0,95	1,00	0,94	0,90	0,86	0,86	0,79	0,81	0,84	0,83	0,79	0,78	0,78	0,78	0,76	0,76	0,79
2005	0,62	0,72	0,68	0,49	0,52	0,72	0,46	0,79	0,82	0,84	0,88	0,94	1,00	0,94	0,88	0,86	0,78	0,81	0,85	0,83	0,80	0,79	0,79	0,78	0,78	0,76	0,77
2006	0,63	0,73	0,68	0,49	0,52	0,72	0,46	0,79	0,82	0,84	0,86	0,94	1,00	0,93	0,89	0,80	0,82	0,85	0,83	0,80	0,78	0,78	0,79	0,77	0,76	0,78	0,78
2007	0,61	0,71	0,67	0,48	0,51	0,71	0,45	0,77	0,80	0,82	0,84	0,86	0,88	0,93	1,00	0,94	0,82	0,81	0,83	0,82	0,79	0,77	0,77	0,77	0,75	0,75	0,78
2008	0,61	0,71	0,67	0,48	0,51	0,71	0,46	0,78	0,80	0,82	0,84	0,86	0,86	0,89	0,94	1,00	0,87	0,86	0,87	0,85	0,81	0,80	0,79	0,79	0,78	0,77	0,78
2009	0,60	0,67	0,63	0,47	0,51	0,66	0,46	0,72	0,73	0,76	0,77	0,79	0,78	0,80	0,82	0,87	1,00	0,89	0,85	0,81	0,77	0,75	0,75	0,75	0,73	0,72	0,74
2010	0,58	0,67	0,63	0,47	0,49	0,67	0,46	0,74	0,75	0,77	0,79	0,81	0,81	0,82	0,81	0,86	0,89	1,00	0,94	0,90	0,82	0,80	0,79	0,79	0,77	0,76	0,77
2011	0,60	0,69	0,66	0,46	0,50	0,69	0,45	0,76	0,78	0,80	0,81	0,84	0,85	0,85	0,83	0,87	0,85	0,94	1,00	0,94	0,87	0,84	0,85	0,82	0,83	0,79	0,80
2012	0,60	0,69	0,65	0,48	0,50	0,69	0,46	0,76	0,78	0,80	0,81	0,83	0,83	0,83	0,82	0,85	0,81	0,90	0,94	1,00	0,92	0,87	0,85	0,84	0,83	0,81	0,82
2013	0,58	0,66	0,62	0,46	0,48	0,66	0,44	0,73	0,75	0,77	0,78	0,79	0,80	0,80	0,79	0,81	0,77	0,82	0,87	0,92	1,00	0,93	0,87	0,81	0,82	0,78	0,79
2014	0,57	0,65	0,61	0,45	0,47	0,65	0,44	0,72	0,74	0,76	0,76	0,78	0,79	0,78	0,77	0,80	0,75	0,80	0,84	0,87	0,93	1,00	0,93	0,85	0,82	0,77	0,78
2015	0,57	0,65	0,61	0,43	0,47	0,65	0,43	0,72	0,74	0,76	0,76	0,78	0,79	0,78	0,77	0,79	0,75	0,79	0,85	0,85	0,87	0,93	1,00	0,91	0,87	0,78	0,79
2016	0,57	0,66	0,62	0,45	0,48	0,66	0,43	0,72	0,74	0,76	0,77	0,78	0,78	0,79	0,77	0,79	0,75	0,79	0,82	0,84	0,81	0,85	0,91	1,00	0,89	0,84	0,81
2017	0,56	0,64	0,60	0,42	0,46	0,64	0,42	0,71	0,73	0,74	0,75	0,76	0,78	0,77	0,75	0,78	0,73	0,77	0,83	0,83	0,82	0,82	0,87	0,89	1,00	0,90	0,86
2018	0,55	0,63	0,60	0,42	0,46	0,63	0,41	0,70	0,72	0,74	0,75	0,76	0,76	0,76	0,75	0,77	0,72	0,76	0,79	0,81	0,82	0,82	0,87	0,84	0,90	1,00	0,92
2019	0,58	0,67	0,63	0,44	0,48	0,66	0,42	0,72	0,74	0,77	0,77	0,79	0,77	0,78	0,78	0,78	0,74	0,77	0,80	0,82	0,79	0,78	0,79	0,81	0,86	0,92	1,00

FB	2012	2013	2014	2015	2016	2017	2018	2019	MAIL.RU	2010	2011	2012	2013	2014	2015	2016	2017	2018	2019
2012	1,00	0,95	0,89	0,85	0,82	0,82	0,81	0,78	2010	1,00	0,89	0,83	0,82	0,83	0,80	0,80	0,76	0,75	0,70
2013	0,95	1,00	0,95	0,90	0,85	0,84	0,83	0,81	2011	0,89	1,00	0,89	0,84	0,83	0,81	0,80	0,75	0,75	0,70
2014	0,89	0,95	1,00	0,96	0,89	0,87	0,85	0,83	2012	0,83	0,89	1,00	0,93	0,87	0,84	0,83	0,78	0,77	0,72
2015	0,85	0,90	0,96	1,00	0,94	0,91	0,87	0,84	2013	0,82	0,84	0,93	1,00	0,93	0,86	0,84	0,79	0,78	0,73
2016	0,82	0,85	0,89	0,94	1,00	0,96	0,90	0,84	2014	0,83	0,83	0,87	0,93	1,00	0,93	0,86	0,81	0,80	0,74
2017	0,82	0,84	0,87	0,91	0,96	1,00	0,95	0,88	2015	0,80	0,81	0,84	0,86	0,93	1,00	0,91	0,82	0,80	0,74
2018	0,81	0,83	0,85	0,87	0,90	0,95	1,00	0,93	2016	0,80	0,80	0,83	0,84	0,86	0,91	1,00	0,88	0,82	0,75
2019	0,78	0,81	0,83	0,84	0,84	0,88	0,93	1,00	2017	0,76	0,75	0,78	0,79	0,81	0,82	0,88	1,00	0,88	0,74
									2018	0,75	0,75	0,77	0,78	0,80	0,80	0,82	0,88	1,00	0,80
									2019	0,70	0,70	0,72	0,73	0,74	0,74	0,75	0,74	0,80	1,00

MTCH	2016	2017	2018	2019	SNAP	2017	2018	2019	LNKD	2011	2012	2013	2014	2015
2016	1,00	0,91	0,88	0,84	2017	1,00	0,96	0,92	2011	1,00	0,96	0,91	0,91	0,89
2017	0,91	1,00	0,96	0,87	2018	0,96	1,00	0,95	2012	0,96	1,00	0,96	0,93	0,90
2018	0,88	0,96	1,00	0,90	2019	0,92	0,95	1,00	2013	0,91	0,96	1,00	0,95	0,91
2019	0,84	0,87	0,90	1,00					2014	0,91	0,93	0,95	1,00	0,96
									2015	0,89	0,90	0,91	0,96	1,00

TWTR	2013	2014	2015	2016	2017	2018	2019	YNDX	2012	2013	2014	2015	2016	2017	2018	2019
2013	1,00	0,95	0,90	0,88	0,87	0,85	0,81	2012	1,00	0,91	0,76	0,64	0,61	0,59	0,61	0,59
2014	0,95	1,00	0,94	0,93	0,91	0,89	0,85	2013	0,91	1,00	0,86	0,67	0,64	0,61	0,63	0,61
2015	0,90	0,94	1,00	0,94	0,91	0,88	0,85	2014	0,76	0,86	1,00	0,81	0,68	0,64	0,65	0,63
2016	0,88	0,93	0,94	1,00	0,97	0,93	0,88	2015	0,64	0,67	0,81	1,00	0,84	0,66	0,65	0,65
2017	0,87	0,91	0,91	0,97	1,00	0,97	0,90	2016	0,61	0,64	0,68	0,84	1,00	0,81	0,66	0,63
2018	0,85	0,89	0,88	0,93	0,97	1,00	0,95	2017	0,59	0,61	0,64	0,66	0,81	1,00	0,85	0,61
2019	0,81	0,85	0,85	0,88	0,90	0,95	1,00	2018	0,61	0,63	0,65	0,65	0,66	0,85	1,00	0,73
								2019	0,59	0,61	0,63	0,65	0,63	0,61	0,73	1,00

Appendix E. Cosine similarity matrixes across companies by years

2010	AMZN	ATVI	EBAY	GOOGL	MAIL	MSFT	RDSA	XING	XOM
AMZN	1,00	0,62	0,60	0,73	0,47	0,67	0,49	0,38	0,56
ATVI	0,62	1,00	0,47	0,59	0,41	0,57	0,40	0,33	0,45
EBAY	0,60	0,47	1,00	0,57	0,38	0,50	0,36	0,30	0,40
GOOGL	0,73	0,59	0,57	1,00	0,44	0,63	0,45	0,35	0,50
MAIL	0,47	0,41	0,38	0,44	1,00	0,42	0,41	0,37	0,38
MSFT	0,67	0,57	0,50	0,63	0,42	1,00	0,43	0,34	0,50
RDSA	0,49	0,40	0,36	0,45	0,41	0,43	1,00	0,37	0,63
XING	0,38	0,33	0,30	0,35	0,37	0,34	0,37	1,00	0,34
XOM	0,56	0,45	0,40	0,50	0,38	0,50	0,63	0,34	1,00

2011	AMZN	ATVI	EBAY	GOOGL	LNKD	MAIL	MSFT	RDSA	XING	XOM
AMZN	1,00	0,66	0,59	0,74	0,72	0,52	0,65	0,49	0,40	0,51
ATVI	0,66	1,00	0,49	0,64	0,63	0,49	0,59	0,43	0,37	0,42
EBAY	0,59	0,49	1,00	0,58	0,53	0,41	0,49	0,36	0,31	0,36
GOOGL	0,74	0,64	0,58	1,00	0,73	0,50	0,63	0,46	0,38	0,45
LNKD	0,72	0,63	0,53	0,73	1,00	0,51	0,62	0,48	0,44	0,46
MAIL	0,52	0,49	0,41	0,50	0,51	1,00	0,47	0,44	0,42	0,38
MSFT	0,65	0,59	0,49	0,63	0,62	0,47	1,00	0,42	0,35	0,45
RDSA	0,49	0,43	0,36	0,46	0,48	0,44	0,42	1,00	0,39	0,63
XING	0,40	0,37	0,31	0,38	0,44	0,42	0,35	0,39	1,00	0,33
XOM	0,51	0,42	0,36	0,45	0,46	0,38	0,45	0,63	0,33	1,00

2012	AMZN	ATVI	EBAY	FB	GOOGL	LNKD	MAIL	MSFT	RDSA	XING	XOM	YNDX
AMZN	1,00	0,37	0,36	0,39	0,40	0,38	0,28	0,36	0,26	0,21	0,29	0,23
ATVI	0,37	1,00	0,48	0,60	0,62	0,59	0,46	0,57	0,40	0,34	0,42	0,37
EBAY	0,36	0,48	1,00	0,56	0,56	0,52	0,40	0,49	0,34	0,29	0,37	0,31
FB	0,39	0,60	0,56	1,00	0,72	0,71	0,53	0,61	0,43	0,37	0,46	0,42
GOOGL	0,40	0,62	0,56	0,72	1,00	0,70	0,48	0,63	0,42	0,36	0,45	0,45
LNKD	0,38	0,59	0,52	0,71	0,70	1,00	0,48	0,59	0,42	0,39	0,44	0,40
MAIL	0,28	0,46	0,40	0,53	0,48	0,48	1,00	0,47	0,43	0,39	0,38	0,42
MSFT	0,36	0,57	0,49	0,61	0,63	0,59	0,47	1,00	0,39	0,33	0,45	0,35
RDSA	0,26	0,40	0,34	0,43	0,42	0,42	0,43	0,39	1,00	0,35	0,61	0,29
XING	0,21	0,34	0,29	0,37	0,36	0,39	0,39	0,33	0,35	1,00	0,31	0,24
XOM	0,29	0,42	0,37	0,46	0,45	0,44	0,38	0,45	0,61	0,31	1,00	0,31
YNDX	0,23	0,37	0,31	0,42	0,45	0,40	0,42	0,35	0,29	0,24	0,31	1,00

2013	AMZN	ATVI	EBAY	FB	GOOGL	LNKD	MAIL	MSFT	RDSA	TWTR	XING	XOM	YNDX
AMZN	1,00	0,63	0,59	0,70	0,72	0,65	0,49	0,65	0,45	0,67	0,37	0,45	0,42
ATVI	0,63	1,00	0,50	0,60	0,64	0,56	0,46	0,55	0,41	0,60	0,34	0,39	0,38
EBAY	0,59	0,50	1,00	0,58	0,57	0,52	0,40	0,50	0,34	0,54	0,30	0,33	0,31
FB	0,70	0,60	0,58	1,00	0,74	0,69	0,52	0,62	0,43	0,80	0,37	0,41	0,41
GOOGL	0,72	0,64	0,57	0,74	1,00	0,67	0,49	0,65	0,43	0,71	0,37	0,42	0,46
LNKD	0,65	0,56	0,52	0,69	0,67	1,00	0,46	0,59	0,40	0,70	0,39	0,38	0,36
MAIL	0,49	0,46	0,40	0,52	0,49	0,46	1,00	0,46	0,44	0,49	0,39	0,34	0,42
MSFT	0,65	0,55	0,50	0,62	0,65	0,59	0,46	1,00	0,39	0,60	0,34	0,40	0,35
RDSA	0,45	0,41	0,34	0,43	0,43	0,40	0,44	0,39	1,00	0,42	0,37	0,63	0,29
TWTR	0,67	0,60	0,54	0,80	0,71	0,70	0,49	0,60	0,42	1,00	0,37	0,39	0,37
XING	0,37	0,34	0,30	0,37	0,37	0,39	0,39	0,34	0,37	0,37	1,00	0,28	0,23
XOM	0,45	0,39	0,33	0,41	0,42	0,38	0,34	0,40	0,63	0,39	0,28	1,00	0,28
YNDX	0,42	0,38	0,31	0,41	0,46	0,36	0,42	0,35	0,29	0,37	0,23	0,28	1,00

2014	AMZN	ATVI	EBAY	FB	GOOGL	LNKD	MAIL	MSFT	RDSA	TWTR	XING	XOM	YNDX
AMZN	1,00	0,64	0,62	0,73	0,76	0,69	0,48	0,64	0,43	0,71	0,36	0,48	0,44
ATVI	0,64	1,00	0,53	0,62	0,67	0,61	0,46	0,56	0,38	0,63	0,33	0,40	0,39
EBAY	0,62	0,53	1,00	0,61	0,63	0,57	0,42	0,52	0,34	0,59	0,30	0,38	0,35
FB	0,73	0,62	0,61	1,00	0,80	0,75	0,52	0,64	0,42	0,82	0,36	0,45	0,44
GOOGL	0,76	0,67	0,63	0,80	1,00	0,74	0,51	0,65	0,43	0,76	0,38	0,46	0,52
LNKD	0,69	0,61	0,57	0,75	0,74	1,00	0,48	0,61	0,42	0,77	0,39	0,44	0,40
MAIL	0,48	0,46	0,42	0,52	0,51	0,48	1,00	0,46	0,40	0,51	0,38	0,34	0,42
MSFT	0,64	0,56	0,52	0,64	0,65	0,61	0,46	1,00	0,37	0,63	0,33	0,43	0,36
RDSA	0,43	0,38	0,34	0,42	0,43	0,42	0,40	0,37	1,00	0,41	0,33	0,53	0,27
TWTR	0,71	0,63	0,59	0,82	0,76	0,77	0,51	0,63	0,41	1,00	0,37	0,43	0,41
XING	0,36	0,33	0,30	0,36	0,38	0,39	0,38	0,33	0,33	0,37	1,00	0,29	0,24
XOM	0,48	0,40	0,38	0,45	0,46	0,44	0,34	0,43	0,53	0,43	0,29	1,00	0,30
YNDX	0,44	0,39	0,35	0,44	0,52	0,40	0,42	0,36	0,27	0,41	0,24	0,30	1,00

2015	AMZN	ATVI	EBAY	FB	GOOGL	LNKD	MAIL	MSFT	RDSA	TWTR	XING	XOM	YNDX
AMZN	1,00	0,64	0,75	0,74	0,67	0,71	0,47	0,64	0,42	0,70	0,32	0,42	0,44
ATVI	0,64	1,00	0,66	0,64	0,60	0,64	0,45	0,58	0,38	0,64	0,29	0,36	0,39
EBAY	0,75	0,66	1,00	0,75	0,70	0,70	0,49	0,62	0,41	0,71	0,33	0,41	0,45
FB	0,74	0,64	0,75	1,00	0,70	0,76	0,52	0,64	0,42	0,80	0,32	0,40	0,44
GOOGL	0,67	0,60	0,70	0,70	1,00	0,66	0,45	0,56	0,38	0,65	0,30	0,36	0,46
LNKD	0,71	0,64	0,70	0,76	0,66	1,00	0,49	0,63	0,42	0,77	0,34	0,40	0,41
MAIL	0,47	0,45	0,49	0,52	0,45	0,49	1,00	0,44	0,39	0,49	0,33	0,29	0,34
MSFT	0,64	0,58	0,62	0,64	0,56	0,63	0,44	1,00	0,35	0,62	0,29	0,39	0,35
RDSA	0,42	0,38	0,41	0,42	0,38	0,42	0,39	0,35	1,00	0,41	0,29	0,57	0,27
TWTR	0,70	0,64	0,71	0,80	0,65	0,77	0,49	0,62	0,41	1,00	0,32	0,37	0,38
XING	0,32	0,29	0,33	0,32	0,30	0,34	0,33	0,29	0,29	0,32	1,00	0,23	0,20
XOM	0,42	0,36	0,41	0,40	0,36	0,40	0,29	0,39	0,57	0,37	0,23	1,00	0,26
YNDX	0,44	0,39	0,45	0,44	0,46	0,41	0,34	0,35	0,27	0,38	0,20	0,26	1,00

2016	AMZN	ATVI	EBAY	FB	GOOGL	MAIL	MSFT	MTCH	RDSA	TWTR	XING	XOM	YNDX
AMZN	1,00	0,67	0,76	0,76	0,72	0,46	0,66	0,59	0,38	0,72	0,31	0,52	0,44
ATVI	0,67	1,00	0,68	0,67	0,68	0,46	0,60	0,55	0,35	0,66	0,30	0,46	0,44
EBAY	0,76	0,68	1,00	0,76	0,74	0,48	0,64	0,59	0,37	0,72	0,32	0,50	0,46
FB	0,76	0,67	0,76	1,00	0,76	0,51	0,66	0,62	0,38	0,81	0,33	0,50	0,45
GOOGL	0,72	0,68	0,74	0,76	1,00	0,46	0,62	0,57	0,35	0,72	0,30	0,47	0,49
MAIL	0,46	0,46	0,48	0,51	0,46	1,00	0,44	0,45	0,35	0,50	0,32	0,35	0,33
MSFT	0,66	0,60	0,64	0,66	0,62	0,44	1,00	0,49	0,34	0,65	0,29	0,47	0,36
MTCH	0,59	0,55	0,59	0,62	0,57	0,45	0,49	1,00	0,35	0,65	0,29	0,40	0,32
RDSA	0,38	0,35	0,37	0,38	0,35	0,35	0,34	0,35	1,00	0,39	0,27	0,56	0,24
TWTR	0,72	0,66	0,72	0,81	0,72	0,50	0,65	0,65	0,39	1,00	0,32	0,47	0,41
XING	0,31	0,30	0,32	0,33	0,30	0,32	0,29	0,29	0,27	0,32	1,00	0,26	0,20
XOM	0,52	0,46	0,50	0,50	0,47	0,35	0,47	0,40	0,56	0,47	0,26	1,00	0,33
YNDX	0,44	0,44	0,46	0,45	0,49	0,33	0,36	0,32	0,24	0			

2018	AMZN	ATVI	EBAY	FB	GOOGL	MAIL	MSFT	MTCH	RDSA	SNAP	TWTR	XING	XOM	YNDX
AMZN	1,00	0,69	0,66	0,77	0,73	0,41	0,69	0,66	0,39	0,69	0,75	0,35	0,52	0,38
ATVI	0,69	1,00	0,55	0,66	0,68	0,40	0,61	0,58	0,33	0,60	0,66	0,31	0,44	0,39
EBAY	0,66	0,55	1,00	0,62	0,59	0,35	0,54	0,55	0,35	0,62	0,61	0,31	0,43	0,28
FB	0,77	0,66	0,62	1,00	0,75	0,44	0,69	0,67	0,37	0,78	0,82	0,34	0,48	0,38
GOOGL	0,73	0,68	0,59	0,75	1,00	0,40	0,64	0,61	0,34	0,67	0,71	0,32	0,46	0,42
MAIL	0,41	0,40	0,35	0,44	0,40	1,00	0,39	0,42	0,29	0,40	0,43	0,32	0,30	0,26
MSFT	0,69	0,61	0,54	0,69	0,64	0,39	1,00	0,54	0,34	0,59	0,68	0,32	0,48	0,32
MTCH	0,66	0,58	0,55	0,67	0,61	0,42	0,54	1,00	0,35	0,67	0,69	0,32	0,43	0,32
RDSA	0,39	0,33	0,35	0,37	0,34	0,29	0,34	0,35	1,00	0,36	0,38	0,29	0,54	0,21
SNAP	0,69	0,60	0,62	0,78	0,67	0,40	0,59	0,67	0,36	1,00	0,73	0,32	0,43	0,35
TWTR	0,75	0,66	0,61	0,82	0,71	0,43	0,68	0,69	0,38	0,73	1,00	0,34	0,47	0,36
XING	0,35	0,31	0,31	0,34	0,32	0,32	0,32	0,32	0,29	0,32	0,34	1,00	0,29	0,20
XOM	0,52	0,44	0,43	0,48	0,46	0,30	0,48	0,43	0,54	0,43	0,47	0,29	1,00	0,28
YNDX	0,38	0,39	0,28	0,38	0,42	0,26	0,32	0,32	0,21	0,35	0,36	0,20	0,28	1,00

2019	AMZN	ATVI	EBAY	FB	GOOGL	MAIL	MSFT	MTCH	RDSA	SNAP	TWTR	XING	XOM	YNDX
AMZN	1,00	0,69	0,75	0,76	0,74	0,50	0,70	0,49	0,39	0,67	0,76	0,41	0,54	0,44
ATVI	0,69	1,00	0,65	0,65	0,68	0,47	0,60	0,43	0,34	0,59	0,66	0,36	0,45	0,43
EBAY	0,75	0,65	1,00	0,69	0,71	0,47	0,62	0,50	0,37	0,68	0,71	0,39	0,49	0,40
FB	0,76	0,65	0,69	1,00	0,73	0,50	0,67	0,50	0,37	0,74	0,77	0,38	0,47	0,38
GOOGL	0,74	0,68	0,71	0,73	1,00	0,48	0,64	0,47	0,35	0,67	0,71	0,38	0,47	0,46
MAIL	0,50	0,47	0,47	0,50	0,48	1,00	0,45	0,42	0,36	0,46	0,49	0,44	0,36	0,40
MSFT	0,70	0,60	0,62	0,67	0,64	0,45	1,00	0,41	0,34	0,58	0,67	0,37	0,46	0,36
MTCH	0,49	0,43	0,50	0,50	0,47	0,42	0,41	1,00	0,29	0,55	0,52	0,32	0,33	0,25
RDSA	0,39	0,34	0,37	0,37	0,35	0,36	0,34	0,29	1,00	0,37	0,39	0,35	0,56	0,23
SNAP	0,67	0,59	0,68	0,74	0,67	0,46	0,58	0,55	0,37	1,00	0,72	0,38	0,44	0,37
TWTR	0,76	0,66	0,71	0,77	0,71	0,49	0,67	0,52	0,39	0,72	1,00	0,42	0,48	0,39
XING	0,41	0,36	0,39	0,38	0,38	0,44	0,37	0,32	0,35	0,38	0,42	1,00	0,34	0,25
XOM	0,54	0,45	0,49	0,47	0,47	0,36	0,46	0,33	0,56	0,44	0,48	0,34	1,00	0,32
YNDX	0,44	0,43	0,40	0,38	0,46	0,40	0,36	0,25	0,23	0,37	0,39	0,25	0,32	1,00

Appendix F. Python code – Chapter 3

```
1. #!/usr/bin/env python
2. # coding: utf-8
3.
4. # # Similarity across companies by years
5.
6. # In[ ]:
7.
8. import os
9. import string
10. import pandas as pd
11. import nltk
12. from csv import writer
13. from difflib import SequenceMatcher
14.
15. def append_list_as_row(file_name, list_of_elem):
16.     # Open file in append mode
17.     with open(file_name, 'a+', newline='') as write_obj:
18.         # Create a writer object from csv module
19.         csv_writer = writer(write_obj)
20.         # Add contents of list as last row in the csv file
21.         csv_writer.writerow(list_of_elem)
22.
23.
24. def similar(a, b):
25.     return SequenceMatcher(None, a, b).ratio()
26.
27. def word_count(str):
28.     counts = dict()
29.     words = str.split()
30.     table = str.maketrans('', '', string.punctuation)
31.     stripped = [w.translate(table) for w in words]
32.     words = [word for word in stripped if word.isalpha()]
33.     from nltk.corpus import stopwords
34.     stop_words = set(stopwords.words('english'))
35.     words = [w for w in words if not w in stop_words]
36.
37.     from nltk.stem.porter import PorterStemmer
38.     porter = PorterStemmer()
39.     words = [porter.stem(word) for word in words]
40.
41.     for word in words:
42.         word = word.lower()
43.         word = word.translate(string.punctuation)
44.         if word in counts:
45.             counts[word] += 1
46.         else:
47.             counts[word] = 1
48.     return counts
49.
50. pd.set_option('display.max_rows', None)
51. directory = 'C:/Users/Desktop/Reports/'
52. results='C:/Users/Desktop/Results/'
53.
54. for year in range(2019,2020):
55.     print(year)
56.     samples =[]
57.     headers = []
58.
59.     for foldername in os.listdir(directory):
```

```

60.         #print(os.listdir(directory+foldername))
61.         for file in os.listdir(directory+foldername):
62.             if file.endswith(str(year)+".txt"):
63.                 headers.append(foldername)
64.                 print(directory+foldername+"/"+file)
65.                 f = open(directory+foldername+"/"+file, encoding="utf8")
66.                 lines = f.read()
67.                 samples.append([foldername+file[0:4],lines])
68.                 continue
69.             else:
70.                 continue
71.         #print(samples[0])
72.         prevsample = ""
73.         print("Length of each sample and similarity with reports of other
companies:")
74.         for sample in samples:
75.             similarity=similar(sample[1],prevsample)
76.             print(sample[0]+" "+str(len(sample[1]))+" "+str(similarity))
77.             prevsample = sample[1]
78.             path = results+str(year)+'stats.csv'
79.             newrow = [sample[0][0:4],str(len(sample[1])),str(similarity)]
80.             append_list_as_row(path,newrow)
81.
82.         count_compare=[]
83.         for sample in samples:
84.             count_compare.append(word_count(sample[1]))
85.         df1=pd.DataFrame(count_compare)
86.         df1=pd.DataFrame.transpose(df1)
87.
88.         from sklearn.feature_extraction.text import TfidfVectorizer
89.         vect = TfidfVectorizer(min_df=1, stop_words="english")
90.         corpus=[]
91.         for sample in samples:
92.             corpus.append(sample[1])
93.         tfidf = vect.fit_transform(corpus)
94.
95.         pairwise_similarity = tfidf * tfidf.T
96.         similarity_df=pd.DataFrame(pairwise_similarity.toarray())
97.
98.         similarity_df.index = headers
99.         similarity_df.columns = headers
100.
101.         similarity_df.to_excel(results+"/"+str(year)+"0_similarity_matrix_disclai
mer.xlsx")
102.         print(similarity_df)
103.
104.
105.         # # Similarity of companys' reports through 1993-2020
106.         #
107.
108.         # In[22]:
109.
110.
111.         import os
112.         import string
113.         import pandas as pd
114.         import nltk
115.         from csv import writer
116.         from difflib import SequenceMatcher
117.
118.         def append_list_as_row(file_name, list_of_elem):
119.             # Open file in append mode

```

```

120.     with open(file_name, 'a+', newline='') as write_obj:
121.         # Create a writer object from csv module
122.         csv_writer = writer(write_obj)
123.         # Add contents of list as last row in the csv file
124.         csv_writer.writerow(list_of_elem)
125.
126.
127. def similar(a, b):
128.     return SequenceMatcher(None, a, b).ratio()
129.
130. def word_count(str):
131.     counts = dict()
132.     words = str.split()
133.     table = str.maketrans('', '', string.punctuation)
134.     stripped = [w.translate(table) for w in words]
135.     words = [word for word in stripped if word.isalpha()]
136.     from nltk.corpus import stopwords
137.     stop_words = set(stopwords.words('english'))
138.     words = [w for w in words if not w in stop_words]
139.
140.     from nltk.stem.porter import PorterStemmer
141.     porter = PorterStemmer()
142.     words = [porter.stem(word) for word in words]
143.
144.     for word in words:
145.         word = word.lower()
146.         word = word.translate(string.punctuation)
147.         if word in counts:
148.             counts[word] += 1
149.         else:
150.             counts[word] = 1
151.     return counts
152.
153.
154. pd.set_option('display.max_rows', None)
155. #directory = 'C:/Users/nikos/Desktop/Facebook - Full reports/'
156. directory = 'C:/Users/nikos/Desktop/Reports/EBAY/'
157.
158. # read files
159. samples = []
160.
161. for filename in os.listdir(directory):
162.     if filename.endswith(".txt"):
163.         f = open(directory+"/"+filename, encoding="utf8")
164.         lines = f.read()
165.         samples.append([filename[0:4],lines])
166.
167.         continue
168.     else:
169.         continue
170.
171.
172.
173. prevsample = ""
174. print("Length of each sample and similarity with previous edition:")
175. for sample in samples:
176.     similarity=similar(sample[1],prevsample)
177.     print(sample[0]+" "+str(len(sample[1]))+" "+str(similarity))
178.     prevsample = sample[1]
179.     path = directory+'stats.csv'
180.     newrow = [sample[0][0:4],str(len(sample[1])),str(similarity)]
181.     append_list_as_row(path,newrow)
182.

```

```

183.
184. count_compare=[]
185. for sample in samples:
186.     count_compare.append(word_count(sample[1]))
187. df1=pd.DataFrame(count_compare)
188. df1=pd.DataFrame.transpose(df1)
189.
190. #Rename Columns
191. names=[]
192. for sample in samples:
193.     names.append(sample[0])
194. df1.columns=names
195.
196. df1['mean'] = df1.mean(numeric_only=True, axis=1)
197. df1=df1.sort_values(by=["mean"],ascending=False)
198.
199. dfplot = df1.head(10)
200.
201. dfplot = dfplot.drop('mean',1)
202.
203. df2 = pd.DataFrame.transpose(dfplot)
204. df2.plot()
205.
206. from sklearn.feature_extraction.text import TfidfVectorizer
207. vect = TfidfVectorizer(min_df=1, stop_words="english")
208. corpus=[]
209. for sample in samples:
210.     corpus.append(sample[1])
211. tfidf = vect.fit_transform(corpus)
212.
213. pairwise_similarity = tfidf * tfidf.T
214. similarity_df=pd.DataFrame(pairwise_similarity.toarray())
215. similarity_df.to_excel(directory+"/0_similarity_matrix_disclaimer.xlsx
    ")
216. print(similarity_df)
217.
218.
219. # # Download data and compute betas by years
220.
221. # In[41]:
222.
223.
224. import numpy as np
225. import matplotlib.pyplot as plt
226. import pandas as pd
227. import datetime
228.
229. from pandas_datareader import data as pdr
230. import yfinance as yf
231.
232. yf.pdr_override()
233.
234. years=[]
235. for i in range(1993,2019):
236.     years.append([str(i)+"-01-01",str(i+1)+"-01-01"])
237.
238.
239. def linreg(x,y):
240.     x = sm.add_constant(x)
241.     model = regression.linear_model.OLS(y,x).fit()
242.     x = x[:,1]
243.     return model.params[0],model.params[1]
244.

```

```

245. betas=[]
246. stocks=["AMZN","ATVI","EBAY","FB","GOOGL","LNKD","MSFT","MTCH","RDSA",
"SNAP","TWTR","XOM"]
247.
248. for stock in stocks:
249.     for year in years:
250.         try:
251.             start = year[0]
252.             end = year[1]
253.             df1 = pdr.get_data_yahoo(stock,start=start,end=end)
254.             df2 = pdr.get_data_yahoo("SPY",start=start,end=end)
255.             return_goog = df1.Close.pct_change()[1:]
256.             return_spy = df2.Close.pct_change()[1:]
257.             #plt.figure(figsize=(20,10))
258.             #return_goog.plot()
259.             #return_spy.plot()
260.             #plt.ylabel("daily returns")
261.             #plt.show()
262.             import statsmodels.api as sm
263.             from statsmodels import regression
264.             X = return_spy.values
265.             Y = return_goog.values
266.             alpha, beta = linreg (X,Y)
267.
268.             yr=int(year[1][:4])-1
269.             betas.append([stock,yr,beta])
270.
271.         except:
272.             pass
273.
274. headers=['Ticker','Year','Beta']
275. betas_df=pd.DataFrame(betas, columns=headers)
276. betas_df.to_excel("C:/Users/nikos/Desktop/betas.xlsx")
277.
278.
279. # # Word dispersion plots (older - lower alpha)
280.
281. # In[2]:
282.
283.
284. import nltk
285. from nltk.corpus import webtext
286. from wordcloud import WordCloud
287. import matplotlib.pyplot as plt
288. from matplotlib import rcParams
289. import os
290. from nltk.probability import FreqDist
291. rcParams.update({'figure.autolayout': True})
292. from nltk.corpus import stopwords
293.
294.
295. # In[7]:
296.
297.
298. folder = "C:/Users/Desktop/Reports/"
299.
300. raw_text = str(open(folder+"YNDX/2019.txt" , encoding="utf-8").read())
301. text = nltk.word_tokenize(raw_text)
302. pos_tagged = nltk.pos_tag(text)
303.
304. wordlist=[]
305. wordlist_raw=[]
306.

```

```

307. #POS_tags=["JJ"]
308. #POS_tags=["VBD"]
309. POS_tags=['NN']
310. #POS_tags=["VB", "MD"]
311. #POS_tags=["MD"]
312.
313.
314. for tag in POS_tags:
315.     words = list(filter(lambda x:x[1]==tag,pos_tagged))
316.     for word in words:
317.         wordlist.append([word[0],tag])
318.         wordlist_raw.append(word[0])
319.
320. sr= stopwords.words('english')
321. clean_wordlist = wordlist_raw[:]
322. for word in wordlist_raw:
323.     if word in stopwords.words('english'):
324.         clean_wordlist.remove(word)
325.
326. freq = nltk.FreqDist(clean_wordlist)
327. freq.plot(30, cumulative=False)
328. #words = ['user', 'data',
'network', 'MAU', 'risk', 'DAU', 'active', 'platform', 'may', 'will', 'could', 'wo
uld']
329. words =
['may', 'could', 'would', 'must', 'might', 'user', 'information', 'data', 'platfo
rm', 'product', 'risk', 'harm', 'incur', 'impact', 'regulatory', 'legal']
330.
331. directory = 'C:/Users/Desktop/Reports/'
332. results='C:/Users/Desktop/Results/'
333. for foldername in os.listdir(directory):
334.     counter = 1/len(os.listdir(directory+foldername))
335.     for file in os.listdir(directory+foldername):
336.         alpha = 1/len(os.listdir(directory+foldername))+counter
337.         if file.endswith(".txt"):
338.             print(foldername,file)
339.             wt_words = webtext.words(directory+foldername+"/"+file) #
Sample data
340.             points = [(x/len(wt_words), y) for x in
range(len(wt_words))
341.                        for y in range(len(words)) if wt_words[x] ==
words[y]]
342.             wt_words = ""
343.             if points:
344.                 x, y = zip(*points)
345.             else:
346.                 x = y = ()
347.             plt.plot(x, y, "rx", scalex=.1,alpha=alpha)
348.             plt.yticks(range(len(words)), words, color="b")
349.             plt.ylim(-1, len(words))
350.             plt.title("Lexical Dispersion Plot "+foldername)
351.             plt.xlabel("Word Offset")
352.             plt.savefig(results+ '/Dispersion_plot-
'+foldername+'.jpeg', figsize=(500,500), dpi=300)
353.             plt.clf()

```

References

1. Ahn, Y.Y., Bagrow, J.P. and Lehmann, S., 2010. Link communities reveal multiscale complexity in networks. *nature*, 466(7307), p.761.
2. Albert, R., Jeong, H. and Barabási, A. 1999. Diameter of the World-Wide Web. *Nature*, 401(6749), pp.130-131.
3. Appelman, B., AOL Inc, 2011. Degrees of separation for handling communications. U.S. Patent 7,949,759.
4. Backstrom, L., Boldi, P., Rosa, M., Ugander, J. and Vigna, S., 2012, June. Four degrees of separation. In *Proceedings of the 4th Annual ACM Web Science Conference* (pp. 33-42). ACM.
5. Barabási, A.L. and Albert, R., 1999. Emergence of scaling in random networks. *Science*, 286(5439), pp.509-512.
6. Barabási, A.L., Albert, R. and Jeong, H., 1999. Mean-field theory for scale-free random networks. *Physica A: Statistical Mechanics and its Applications*, 272(1-2), pp.173-187.
7. Barabási, A. 2016. *Network science*. Cambridge: Cambridge University Press.
8. Blanchard, P., Chang, C.H. and Krüger, T., 2003, December. Epidemic thresholds on scale-free graphs: The interplay between exponent and preferential choice. In *Annales Henri Poincaré* (Vol. 4, No. 2, pp. 957-970). Birkhäuser-Verlag.
9. Boguá, M., Pastor-Satorras, R. and Vespignani, A., 2003. Epidemic spreading in complex networks with degree correlations. In *Statistical mechanics of complex networks* (pp. 127-147). Springer, Berlin, Heidelberg.
10. Boguá, M. and Pastor-Satorras, R., 2002. Epidemic spreading in correlated complex networks. *Physical Review E*, 66(4), p.047104.
11. Boguá, M., Pastor-Satorras, R. and Vespignani, A., 2003. Absence of epidemic threshold in scale-free networks with degree correlations. *Physical review letters*, 90(2), p.028701.
12. Boguá, M., Pastor-Satorras, R. and Vespignani, A., 2003. Epidemic spreading in complex networks with degree correlations. In *Statistical mechanics of complex networks* (pp. 127-147). Springer, Berlin, Heidelberg.
13. Cadwalladr, C. and Graham-Harrison, E., 2018. Revealed: 50 million Facebook profiles harvested for Cambridge Analytica in major data breach. *The guardian*, 17, p.22
14. Catanese, S.A., De Meo, P., Ferrara, E., Fiumara, G. and Provetti, A., 2011, May. Crawling facebook for social network analysis purposes. In *Proceedings of the international conference on web intelligence, mining and semantics* (p. 52). ACM.

15. Damodaran Online. Pages.stern.nyu.edu. 2020. [online] Available at: <<http://pages.stern.nyu.edu/~adamodar/>> [Accessed 12 June 2020].
16. Dijkstra, E.W., 1959. A note on two problems in connexion with graphs. *Numerische mathematik*, 1(1), pp.269-271.
17. Doreian, P. and Conti, N., 2012. Social context, spatial structure and social network structure. *Social networks*, 34(1), pp.32-46.
18. Dunbar, R.I., 1998. The social brain hypothesis. *Evolutionary Anthropology: Issues, News, and Reviews: Issues, News, and Reviews*, 6(5), pp.178-190.
19. Edunov, S., Diuk, C., Filiz, I.O., Bhagat, S. and Burke, M., 2016. Three and a half degrees of separation. *Research at Facebook*.
20. Ellison, N., Steinfield, C. and Lampe, C., 2006. Spatially bounded online social networks and social capital. *International Communication Association*, 36(1-37).
21. Erdős P., Rényi A. 1959. On random graphs', *Math Debrecen* 6, 290-297.
22. Evans T.S., Lambiotte R. 2009. Line graphs, link partitions, and overlapping communities. *Physical Review*.
23. Facebook, Inc. , 2013. Generating business results on Facebook. Available at: <https://newsfeed.cz/wp-content/uploads/Generating-business-results-on-Facebook.pdf> (Accessed: 23 February 2020).
24. Facebook, Inc. , 2020. Annual report 2019. Available at: <https://investor.fb.com/financials/default.aspx> (Accessed: 12 February 2020).
25. Feld, S.L., 1991. Why your friends have more friends than you do. *American Journal of Sociology*, 96(6), pp.1464-1477.
26. Fine, B., 2010. *Theories of social capital: Researchers behaving badly*. London: Pluto Press.
27. Freeman, L.C. and Thompson, C.R., 1989. Estimating acquaintanceship volume. *The small world*, pp.147-158.
28. Gilbert E. N. 1959. Random graphs. *The Annals of Mathematical Statistics*, 30:1141-1144.
29. Girvan, M. and Newman, M.E., 2002. Community structure in social and biological networks. *Proceedings of the national academy of sciences*, 99(12), pp.7821-7826.
30. Gjoka, M., Kurant, M., Butts, C.T. and Markopoulou, A., 2010, March. Walking in facebook: A case study of unbiased sampling of osns. In *2010 Proceedings IEEE Infocom* (pp. 1-9). Ieee.
31. Gneiser, M., Heidemann, J., Klier, M., Landherr, A., & Probst, F. (2012). Valuation of online social networks taking into account users' interconnectedness. *Information Systems and E-Business Management*, 10(1), 61–84.

32. Grassberger, P., 1983. On the critical behavior of the general epidemic process and dynamical percolation. *Mathematical Biosciences*, 63(2), pp.157-172.
33. Gonçalves, B., Perra, N. and Vespignani, A., 2011. Modeling users' activity on twitter networks: Validation of dunbar's number. *PloS one*, 6(8), p.e22656.
34. Hachuyan A. 2017. Data analysis for detecting opinion leaders. Rusbase. Available at <https://rb.ru/list/socialdatahub-on-big-data-conference/> (Accessed: 20.05.2020).
35. Hagberg, A., Swart, P. and S Chult, D., 2008. Exploring network structure, dynamics, and function using NetworkX (No. LA-UR-08-05495; LA-UR-08-5495). Los Alamos National Lab.(LANL), Los Alamos, NM (United StatesHanifan, L.J., 1916. The rural school community center. *The Annals of the American Academy of Political and Social Science*, 67(1), pp.130-138.
36. Hatmaker, T. 2018. Facebook will cut off access to third party data for ad targeting. TechCrunch, 29 March 2018 Available at: <https://techcrunch.com/2018/03/28/facebook-will-cut-off-access-to-third-party-data-for-ad-targeting/> (Accessed: 15.02.2020).
37. OECD Insights, 2007. Human Capital. Wie Wissen unser Leben bestimmt. Online verfügbar unter <http://www.oecd.org/berlin/publikationen/humankapitalwiewissenunserlebenbestimmt.htm> [Stand: 07.08. 2014].
38. Irvine, M. 2019. Facebook Ad Benchmarks for YOUR Industry. Wordstream.com. Available at: <https://www.wordstream.com/blog/ws/2017/02/28/facebook-advertising-benchmarks> (Accessed 23 Feb. 2020).
39. Irvine, M. 2020 Google Ads Benchmarks for YOUR Industry. Wordstream.com. Available at: <https://www.wordstream.com/blog/ws/2016/02/29/google-adwords-industry-benchmarks> (Accessed 23 Feb. 2020).
40. Katz, M. L., & Shapiro, C. 1985. Network externalities, competition, and compatibility. *The American economic review*, 75(3), 424-440.
41. Kim E.D.J. and Keng B.J.L., MARKETWIRE LP, 2015. Systems and Methods for Determining Influencers in a Social Data Network. U.S. Patent Application 14/522,471.
42. Kleinfield J., 2002. Could it be a big world after all? The six degrees of separation myth. *Society*, April, 12, pp.5-2.
43. Kosterev D.N., Taylor, C.W. and Mittelstadt, W.A., 1999. Model validation for the August 10, 1996 WSCC system outage. *IEEE transactions on power systems*, 14(3), pp.967-979.
44. Kurucu G., 2007. Negative network externalities in two-sided markets: a competition approach.
45. Jin, Y., Shobowale, S., Koehler, J. and Case, H., 2012. The incremental reach and cost efficiency of online video ads over tv ads.

46. Joo, Y. H., Kim, Y., & Yang, S. J. 2011. Valuing customers for social network services. *Journal of Business Research*, 64(11), 1239-1244.
47. Johanson, C., Altaba Inc, 2009. Ranking content based on social network connection strengths. U.S. Patent Application 11/851,629.
48. Kessing, S.G. and Nuscheler, R., 2006. Monopoly pricing with negative network effects: The case of vaccines. *European Economic Review*, 50(4), pp.1061-1069.
49. Lawrence, S. and Giles, C.L., 1999. Accessibility of information on the web. *Nature*, 400(6740), p.107.
50. Leskovec, J., Lang, K.J., Dasgupta, A. and Mahoney, M.W., 2009. Community structure in large networks: Natural cluster sizes and the absence of large well-defined clusters. *Internet Mathematics*, 6(1), pp.29-123.
51. Lunden I. 2013. Facebook launches partner categories, 500+ generic profiles to target ads better, with data from Datalogix, Epsilon, Acxiom. *TechCrunch*, 10 April 2013. Available at: <http://techcrunch.com/2013/04/10/facebook-launches-partner-categories-500-profiles-to-target-ads-better-on-mobile-and-desktop-using-data-from-datalogix-epsilon-and-acxiom/> (Accessed: 12 February 2020).
52. McAuley J. and Leskovec J. Learning to Discover Social Circles in Ego Networks. NIPS, 2012.
53. Microsoft Corporation, 2020. Annual report 2019. Available at: <https://www.microsoft.com/en-us/Investor/sec-filings.aspx> (Accessed: 12 February 2020).
54. Miller, K., Istvan, Z., Joel, C.S. and Peter, L., EoN (Epidemics on Networks): a fast, flexible Python package for simulation, analytic approximation, and analysis of epidemics on networks. *Journal of Open Source Software*, 4(44), 1731
55. Moreno, Y. and Vazquez, A., 2003. Disease spreading in structured scale-free networks. *The European Physical Journal B-Condensed Matter and Complex Systems*, 31(2), pp.265-271.
56. MSU. Moscow State University official website. Available at: <http://msu.ru/> (Accessed: 21 February 2020).
57. Myers, S.A., Sharma, A., Gupta, P. and Lin, J., 2014, April. Information network or social network?: the structure of the twitter follow graph. In *Proceedings of the 23rd International Conference on World Wide Web* (pp. 493-498). ACM.
58. Newman, M.E., 2002. Spread of epidemic disease on networks. *Physical review E*, 66(1), p.016128.
59. Newman, M.E., 2003. The structure and function of complex networks. *SIAM review*, 45(2), pp.167-256.

60. Newman, M.E. and Girvan, M., 2004. Finding and evaluating community structure in networks. *Physical review E*, 69(2), p.026113.
61. Nltk.org. 2020. Natural Language Toolkit — NLTK 3.5 Documentation. (Online) Available at: <https://www.nltk.org/> (Accessed 7 June 2020).
62. Liu, J., Pan, Y., Hu, Q. and Li, A., 2019, June. Navigating a Shortest Path with High Probability in Massive Complex Networks. In *International Symposium on Experimental Algorithms* (pp. 82-97). Springer, Cham.
63. Lőrincz, L., Koltai, J., Győr, A. F., & Takács, K. (2019). Collapse of an online social network: Burning social capital to create it? *Social Networks*, 57, 43–53. <https://doi.org/10.1016/j.socnet.2018.11.004>
64. Milgram, S., 1967. The small world problem. *Psychology today*, 2(1), pp.60-67.
65. Osterwalder, A. and Pigneur, Y., 2010. *Business model generation: a handbook for visionaries, game changers, and challengers*. John Wiley & Sons.
66. Pergelova A. , Prior D. & Rialp J. (2010) Assessing Advertising Efficiency, *Journal of Advertising*, 39:3, 39-54, DOI: 10.2753/JOA0091-3367390303
67. Pool S.I., Kochen, M., 1978. Contacts and influence. *Social networks*, 1(1), pp.5-51.
68. Portes, A., 1998. Social capital: Its origins and applications in modern sociology. *Annual review of sociology*, 24(1), pp.1-24.
69. Porter M., *Competitive Advantage: Creating and sustaining superior Performance* (New York: Free Press, 1985,1995).
70. Putnam, R.D., 2000. *Bowling alone: The collapse and revival of American community*. Simon and schuster.
71. Ravasz, E., Somera, A.L., Mongru, D.A., Oltvai, Z.N. and Barabási, A.L., 2002. Hierarchical organization of modularity in metabolic networks. *science*, 297(5586), pp.1551-1555.
72. Resnick, P., 2001. Beyond bowling together: Sociotechnical capital. *HCI in the New Millennium*, 77, pp.247-272.
73. Rosenthal, H.L., 1960. *Acquaintances and contacts of Franklin Roosevelt: the first 86 days of 1934* (Doctoral dissertation, Massachusetts Institute of Technology).
74. Sander, L.M., Warren, C.P., Sokolov, I.M., Simon, C. and Koopman, J., 2002. Percolation on heterogeneous networks as a model for epidemics. *Mathematical biosciences*, 180(1-2), pp.293-305.
75. Schmittlein, D.C., Morrison, D.G. and Colombo, R., 1987. Counting your customers: Who are they and what will they do next?. *Management science*, 33(1), pp.1-24.
76. Schnettler, S., 2009. A small world on feet of clay? A comparison of empirical small-world studies against best-practice criteria. *Social Networks*, 31(3), pp.179-189.

77. Shapiro, C., Carl, S. and Varian, H.R., 1998. Information rules: a strategic guide to the network economy. Harvard Business Press.
78. Stella, M., Ferrara, E. and De Domenico, M., 2018. Bots increase exposure to negative and inflammatory content in online social systems. *Proceedings of the National Academy of Sciences*, 115(49), pp.12435-12440.
79. Travers, J. and Milgram, S., 1969. An exploratory study of the small world problem. *Sociometry*, 32, pp.425-43.
80. Ugander, J., Karrer, B., Backstrom, L., & Marlow, C. 2011. The anatomy of the facebook social graph. arXiv preprint arXiv:1111.4503.
81. Upwork, Inc. , 2020. Annual report 2019. Available at: <https://investors.upwork.com/node/6731/html> (Accessed: 28 February 2020).
82. Vázquez, A. and Moreno, Y., 2003. Resilience to damage of graphs with degree correlations. *Physical Review E*, 67(1), p.015101.
83. VK. V Kontakte LLC. available at <https://vk.com> (Accessed: 21 February 2020)
84. Wellman, B., Haase, A.Q., Witte, J. and Hampton, K., 2001. Does the Internet increase, decrease, or supplement social capital? Social networks, participation, and community commitment. *American behavioral scientist*, 45(3), pp.436-455.
85. Watanabe, M. and Suzumura, T., 2013, May. How social network is evolving?: a preliminary study on billion-scale twitter network. In *Proceedings of the 22nd International Conference on World Wide Web* (pp. 531-534). ACM.
86. Wu, T. 2016. The attention merchants: the epic scramble to get inside our heads.
87. XING SE, 2019. Annual report 2018. Available at: <https://www.new-work.se/en/investor-relations/publications/> (Accessed: 12 February 2020).
88. Yook, S.H., Jeong, H. and Barabási, A.L., 2002. Modeling the Internet's large-scale topology. *Proceedings of the National Academy of Sciences*, 99(21), pp.13382-13386.
89. Zhao, J., Wu, J., Liu, G., Tao, D., Xu, K. and Liu, C., 2014. Being rational or aggressive? A revisit to Dunbar' s number in online social networks. *Neurocomputing*, 142, pp.343-353.
90. Zhang, X.Z., Liu, J.J. and Xu, Z.W., 2015. Tencent and Facebook data validate Metcalfe's law. *Journal of Computer Science and Technology*, 30(2), pp.246-251.
91. Zhou, D.T., Zhou, T.T. and Zhou, A.H., 2015. Method and system for social credit scoring. U.S. Patent 9,009,166.
92. Zhu, X.M. and Lunt, C., Friendster Inc, 2008. Compatibility scoring of users in a social network. U.S. Patent 7,451,161.