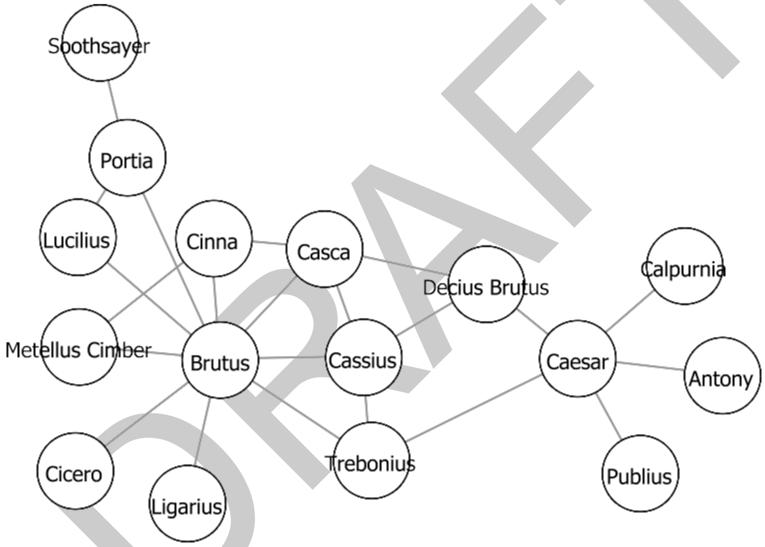
DYNAMIC NETWORK ANALYSIS

BY

KATHLEEN M. CARLEY



DRAFT

Draft
January 21, 2013

Center for Computational Analysis
of Social and Organizational Systems
Carnegie Mellon University, Pittsburgh, USA

Many people were involved in the development of this book. Different chapters of the book were created with the support of Matt deReno, Jamie Olson, Terrill Frantz, Jana Diesner, Brian Hirshman, George Davis, Peter Landwehr, Jeff Reminga, Ju-Sung Lee, and Hope Armstrong.

DRAFT

Preface

Everything is connected! From small groups to economic markets to global societies – interactions among people, organizations, technology, and policies lead to complex systems. These connected systems cannot be described with simple equations–they need to be articulated as networks. *Social Network Analysis (SNA)* offers a wide variety of tools to analyze complex connected systems. The ability to store and work with network data in a digital environment has enabled a multiplicity of new analytic methods for networks. Layout algorithms help analysts create attractive and compelling renderings of data. More sophisticated techniques for detecting significant groups and agents can be applied to larger networks than ever before. These developments and others have changed our way of perceiving and analyzing networks in the world, and they are the ground layer for future understanding of how networks function.

Dynamic Network Analysis (DNA) brings network reasoning to a new level by going beyond the social connections. By adding organizations, knowledge, tasks, locations, beliefs, etc. to the network data and analyzing all these information together as well as the change over time offers new insights into complex socio-cultural systems. This is a teaching book for learning DNA. It is intended for students in all majors as well as for non-academia people who want to analyze networks. The book is targeted to an audience that has little or no experience with network analysis. For the advanced reader, the book serves as a reference book, as it offers an extensive glossary and a collection of analytical network algorithms. Readers who are familiar with SNA can learn how to extend the scope of analysis beyond people to multi-modal networks of people, organizations, task, resources, knowledge, events, and locations.

This book is not a manual for a specific software program, but rather an introduction to DNA. Nevertheless, every researcher and every analyst needs software to perform her or his research. We use ORA. ORA is a powerful software tool to handle and analyze dynamic networks and is developed by

DRAFT

the Center for Computational Analysis of Social and Organizational Systems (CASOS) at Carnegie Mellon University. To get the most out of learning dynamic network analysis with this book, we recommend using ORA, too.

The book is organized as follows. The first three chapters are introductions to DNA. We suggest reading those chapters first. Chapter 1 is a general introduction and it also gives a plot overview of *The Tragedy of Julius Caesar*, a play by William Shakespeare which is primarily used in this book to introduce different aspects of network analysis. Chapter 2 introduces the topic of SNA. Chapter 3 expands people networks to meta-networks by including other entity classes and covers the different methods that can be followed when using networks consisting of two or more entity classes. Chapter 4 shows you how to analyze groups in networks. Space (chapter 5) and time (chapter 6) are covered in the following chapters. Chapter 7 discusses simulation of network data and chapter 8 presents various aspects of network text analysis. Finally, chapter 9 discusses future directions of DNA. The last part of the book covers a structured collection of many methods and algorithms for DNA. We do not discuss the algorithms of these measures in detail during the course of the chapters, but they are defined at the end of the book.

At the end of every chapter you will find problem sets. You can use them to test your knowledge about the topics that are presented in each chapter. We do not offer solutions to these problem sets because often the answer is not trivially ”yes” or ”no,” but you will be able to find all the information you need to solve the questions in the book itself.

DRAFT

Contents

1 The Essence of Network Analysis 15

1.1 Network analysis beyond the social graph 16

1.1.1 Communication as a function of distance 17

1.1.2 Co-word analysis of invisible colleges 18

1.1.3 An acquaintance process 19

1.2 Dynamic Network Analysis as answer 19

1.2.1 Who can use DNA? 22

1.2.2 More than "who you know" 23

1.3 Caesar, Brutus, and co. 26

1.3.1 The plot of the tragedy 27

1.3.2 Saving Julius Caesar? 31

1.4 Problem set 32

2 Analyzing Social Networks 35

2.1 Units of interest 35

2.1.1 Entities: Friends, romans, and countrymen 35

2.1.2 Relations: To love and to hate 37

2.2 More definitions 39

2.3 The structure of human connections 41

2.3.1 Networks on personal level 41

2.3.2 Networks on group and society level 43

2.4 Analyzing social networks 47

2.4.1 Visualizing networks 47

DRAFT

2.4.2 Networks over time 48
2.4.3 Identifying important agents 50
2.5 Problem set 54

3 Meta-Networks 57
3.1 More than *who*: Additional entity classes 59
3.1.1 Skills in the roman empire 62
3.1.2 Tasks that drive the tragedy 63
3.1.3 Where everything takes place 64
3.1.4 Events as corner posts of the story 66
3.1.5 Some comments on node classes 67
3.2 Adding it all together – the meta-network 69
3.3 Concepts for two entity classes 70
3.3.1 Quantity 70
3.3.2 Variance 74
3.3.3 Correlation 75
3.3.4 Specialization 77
3.4 Concepts for three and more entity classes 79
3.4.1 Quantity 81
3.4.2 Coherence 81
3.4.3 Substitution 82
3.4.4 Control 82
3.5 Problem set 84

4 Finding Groups 87
4.1 Interesting patterns 90
4.2 Grouping Methods 91
4.2.1 Newman grouping 91
4.2.2 CONCOR grouping 92
4.2.3 Johnson grouping 93
4.2.4 Fuzzy Grouping 93
4.2.5 Block-Modeling 93
4.2.6 Other approaches 94

DRAFT

4.3 Groups in meta-networks 94
4.4 Problem set 94

5 Spatially Embedded Networks 97
5.1 Propinquity – Those close by form a tie 98
5.2 GIS, shape-files, and Co. 98
5.3 Spatial visualizations 98
5.4 Spatial centralities 98

6 Temporal Networks 99
6.1 Networks over time 99
6.1.1 Creating networks over time 99
6.1.2 Levels of aggregation 100
6.2 Trails 102
6.3 Measuring change 104
6.3.1 Levels of comparison 104
6.3.2 Network distances 105
6.3.3 Correlation of networks and its problems 107
6.3.4 QAP/MRQAP 108
6.3.5 Exponential Random Graph (p^*) Models 110
6.4 Detecting change 111
6.4.1 Shewhart's Chart 112
6.4.2 Cumulative Sum (CUSUM) 115
6.5 Periodicities 116
6.6 Problem set 117

7 Network Evolution and Diffusion 121
7.1 Diffusion of apples, ideas and beliefs 121
7.1.1 Random networks and stylized networks 121
7.1.2 Diffusion of innovation 121
7.1.3 Epidemic concepts 121
7.2 Agent-based dynamic-network computer simulations 121
7.2.1 Models 121

DRAFT

7.2.2 Models for diffusion processes ... 121
7.3 Evolution of networks ... 121

8 Extracting Networks from Texts 123
8.1 Analyzing texts ... 124
8.1.1 Content analysis ... 124
8.1.2 tf*idf ... 124
8.2 Text processing ... 124
8.2.1 Deletion ... 124
8.2.2 Thesauri ... 124
8.2.3 Concept lists ... 124
8.2.4 Bi-grams ... 124
8.2.5 Stemming ... 124
8.3 From texts to networks ... 124
8.3.1 Keyword in context ... 124
8.3.2 Windowing ... 124
8.3.3 Extracting meta-networks from texts ... 124

9 The Future of Dynamic Network Analysis 125

Appendix A SNA Measures Glossary 127
A.1 Notations ... 128
A.1.1 Node Classes ... 128
A.1.2 Matrices ... 128
A.2 Standard network measures ... 128
A.2.1 Degree Centrality ... 128
A.2.2 Closeness Centrality ... 128
A.2.3 Betweenness Centrality ... 129
A.2.4 Eigenvector Centrality ... 130
A.2.5 Clustering Coefficient ... 130
A.3 Grouping algorithms ... 131
A.4 Change measures ... 131
A.5 Network text algorithms ... 131

DRAFT

Appendix B Two-Mode Network Measures 133

B.1 Quantity 133

B.1.1 Degree 133

B.1.2 Load 133

B.2 Variance 134

B.2.1 Centralization 134

B.2.2 Diversity 134

B.3 Correlation 135

B.3.1 Similarity 135

B.3.2 Distinctiveness 136

B.3.3 Resemblance 136

B.3.4 Expertise 136

B.4 Specialization 137

B.4.1 Exclusivity 137

B.4.2 Redundancy 137

B.4.3 Access 138

Appendix C Multi-Mode Network Measures 139

C.1 Quantity 139

C.1.1 Degree 139

C.1.2 Load 139

C.2 Coherence 140

C.2.1 Congruence 140

C.2.2 Needs 141

C.2.3 Waste 142

C.2.4 Performance 143

C.2.5 Workload 143

C.2.6 Negotiation 144

C.3 Substitution 144

C.3.1 Availability 144

C.3.2 Reuse 145

C.4 Control 145

DRAFT

C.4.1 Demand 145
C.4.2 Awareness 147

Appendix D Julius Caesar Data **149**
D.1 Interactions of Agents 149
D.2 Agents and Their Connections 149
D.3 Other Networks 149

DRAFT

List of Figures

1.1 Distance and communication 17
1.2 The genealogical tree of dynamic network analysis 24

2.1 Triads of Heider’s balance theory 43
2.2 Hierarchical organization of nodes in a personal network 45
2.3 Network visualization of the first act of Julius Caesar 48
2.4 Networks of five acts of Julius Caesar 49
2.5 A simple network to illustrate different aspects of centrality . . 51
2.6 Degree centrality of six agents over the course of the tragedy . 52
2.7 Change of betweenness centrality over time 53

3.1 Agent and Location network 65
3.2 Event x Event network of Events following each other 67
3.3 Conceptual connections between the four main node classes . . 80

4.1 Network visualization of the first act of Julius Caesar 89
4.2 Example for line betweenness centrality 92

6.1 Aggregation of temporal data 101
6.2 Aggregation of temporal network data 102
6.3 Hamming distance of two networks 106
6.4 Quadratic Assignment Procedure 109
6.5 Shewhart chart for degree centralization of e-mail data 114
6.6 Cumulative sum chart 116

DRAFT

D.1 Agent by Knowledge link matrix 150
D.2 Agent by Location link matrix 150

DRAFT

List of Tables

2.1 Entities (cast) of characters in Julius Caesar (*whos*) 36

2.2 Social network matrix of the first act of Julius Caesar 38

3.1 Knowledge list in Julius Caesar 63

3.2 Task list in Julius Caesar 64

3.3 Locations where the tragedy takes place 64

3.4 Events of the tragedy 66

3.5 Meta-network matrix with all possible 55 networks 71

3.6 Measure concepts for two entity classes 72

3.7 Load and Density of different networks 73

3.8 Centralization and Diversity of Degree Centrality 75

3.9 Correlation measures of Agents in Caesar's empire 77

3.10 Measure concepts for three and more entity classes 79

4.1 Block modeling the social network matrix of the first act of Julius Caesar 94

A.1 Node Classes 128

A.2 Matrix notations 129

DRAFT

DRAFT

Chapter 1

The Essence of Network Analysis

DRAFT

Traditionally, network analysis has focused on the social network–*who* interacts with *whom*. Most classic measures were developed from research on such networks and were meant to be interpreted in a social context. For example, a researcher might survey fraternity members about who they consider to be their friends (Newcomb, 1961). The most popular individual would show up as the *Agent* with the most in-links.

Unlike a conventional social network, a dynamic network bundles together a variety of networks between different types of entities into a meta-network. These different networks include the social network mentioned above, as well as the membership networks. The membership networks are the relationships of the students to their different fraternities, and the inter-organizational network—and the relationships of the fraternities to each other. Humans are not simply situated within one social network but rather a vast sea of overlapping networks of different types. An analyst working with a dynamic network attempts to choose a particular set from this ocean of relationships that is most relevant to their work, bundles them together, and then tries to incorporate and layer them into their analysis. The analyst needs to understand the context in which a social network operates. Thus, rather than asking just *who* do you know, dynamic network analysis supports asking additional context questions such as: How does *who* you know impact *what* you know? *What* you do? *Where* you do it? And of course, networks change over time. In other words, networks of interaction are now embedded in complex meta-networks that link *who*, *what*, *how*, and *why* through time and space. Dynamic Network Analysis (DNA) is the study of these complex networks, generally from a quantitative perspective. *Agent*-based simulation is often used to forecast change and explore variations in the networks over time. In this book, we will move from the basics of Social Network Analysis (SNA) to the more detailed DNA.

DNA can be applied in a wide number of settings. Gaining an understanding of the structure of Al-Qaeda is critical in fighting the war on terror and could help prevent future events such as another September 11 attack. Possessing a true ecological map of a food chain will help keep environments stable (Johnson et al., 2001). Because of limited resources, understanding the varied shipping lanes merchant marine vessels traverse as they conduct international trade is vital to protecting ports of call (Davis and Carley, 2007). Understanding how a network of satellites is connected to various locations around the world is critical for a global company's bottom line. A financial network, such as those that enabled the fraud at Enron to destroy the entire company and make a lifetime's retirement fund disappear in a day, is also fertile territory for DNA.

Networks surround us and pervade our interactions. Your co-workers, your food supply, and even your own body can be construed as networks. You cannot go through life without belonging to a network of some sort. You can even belong to vastly different multitudes of networks which are all interconnected to other sorts of networks that you may or may not have a clue as to their very existence. Even if you are not associated with one particular network, you can still be defined by your isolation from it.

So how can we use an understanding of the relations among *who*, *what*, *where*, *how*, *why* and *when* to analyze how complex systems such as food chains plotted to blow up a U.S. Embassy in Tanzania, the plot to murder Julius Caesar, the performance of public health organizations, the merger of companies, and so on? The answer lies in the science of DNA, a robust approach to network analysis.

DRAFT

1.1 Network analysis beyond the social graph

It is not all about who knows whom. A lot of other factors are important when it comes to analyzing networks. And even if we are interested in who knows whom, looking at where the *Agents* are located, having a closer look at not just who talks to whom but also what the people are talking about, or analyzing the interaction patterns over time, may help us to gain more and better insights into network dynamics. To give you a better impression of

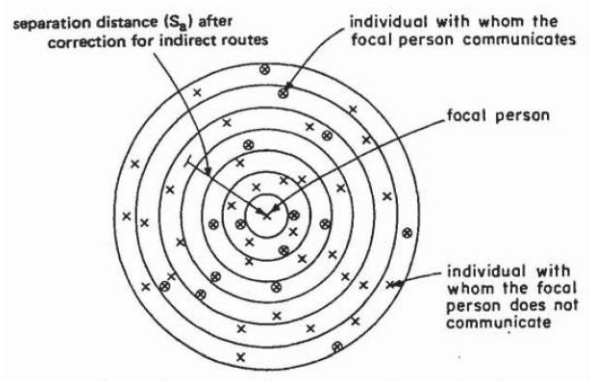

separation distance (S_s) after correction for indirect routes
individual with whom the focal person communicates
focal person
individual with whom the focal person does not communicate

Figure 1.1: Distance and communication in a research laboratory (Allen and Fustfeld, 1975).

DRAFT

what we are talking about, you can find three examples of studies, in which researchers analyzed networks beyond the single social graph.

1.1.1 Communication as a function of distance

Allen and Fustfeld (1975) surveyed people in seven research laboratories. Based on a list of all colleagues, the interviewee where asked with whom they communicate “about technical or scientific matters” (Allen and Fustfeld, 1975, p. 154) at least once a week. In addition, the physical distances between the desks of all employees were measured. For every scientist a radial distance network was created with the person of interest as the focal node in the middle and a circle for every 3 meters (10 feet) forming distance groups (see Figure 1.1). Furthermore, the proportion of communication partners was calculated for every distance group.

In their analysis, Allen and Fusfeld were able to show that the probability of communication follows an exponential decay[1] as a function of distance. This was not a very surprising result for the authors. More astonishing was the rate of the decay. Allen and Fusfeld figured out that the probability of weekly communication is almost zero for any distance between two scientists having their desks more than 25–30 meters. Even more, they show the results of intervention experiments in which the communication between groups of people could be increased significantly just by moving their offices closer to each other. Allen and Fustfeld (1975) were able to show strong evidence that

[1] The authors fit the decay with a hyperbola curve.

“communication is influenced by the physical, architectural arrangement” of the working environment. Similar results were shown in subsequent studies in the following years. A recent study including a review on spatial dimensions of social networks is provided by Sailer and McCulloh (2012). All studies about social networks and geographical proximity or distance have one result in common: Space matters! We will talk more about geo-spatial networks in chapter 5.

1.1.2 Co-word analysis of invisible colleges

Networks of scientists in a particular field who collaborate for publications and grant proposals—independently from their institutional affiliation—are called *Invisible colleges* (Crane, 1972). Lievrouw et al. (1987) were interested in possible invisible colleges in a special sub-field of biomedical science. Lievrouw and her colleagues asked the scientists in the field about their collaboration partners. They looked for the *intellectual* structure of the field by analyzing co-citation as well as common use of content in scientific articles. The content analysis was accomplished by making use of index terms that were pre-assigned by a team of professional indexers at the NIH. These indexers had used a thesaurus of 13,000 words and had assigned on average 15 terms to every grant. Lievrouw et al. (1987) showed in their work that

> “. . . there is indeed a distinction between the communication structure, or social network, among scientists, and the actual content of the work in which they engage.” (Lievrouw et al., 1987, p. 245)

The authors were able to reject the assumption that “social structures in science somehow reflect or represent the intellectual structure of the research specialty” (Lievrouw et al., 1987, p. 245). Even most scientists in the analyzed field knew each other and had many collaboration overlap, the content analysis of their work showed different groups of specialization. From the perspective of raising money for grants, these groups were competitors in a fight for constraint research *Resources*. In a nutshell, Lievrouw et al. (1987) showed in their research that analyzing the content of written text can result in broader insights into the connections between people than just looking at their social interaction ties. Networks that are created from large amounts of text data are a very important aspect of DNA since an enormous amount of text is created every day by journalists describing incidents around the world and by billions of Internet and in particular Social Media users. We discuss the different aspects of network text analysis in chapter 8.

DRAFT

1.1.3 An acquaintance process

Analyzing change in networks over time is another very essential task in DNA. In chapter 6 of this book, you will learn a lot about networks over time and how to measure and detect change in these dynamic networks. At this point, we tell you about the most famous analysis of networks over time in Social Network literature. Theodore Newcomb (1961) was interested in the dynamics that drive the formation of friendship ties—consequently, the resulting book is titles *The Acquaintance Process*. To gather the data that was required for his research, Newcomb recruited 17 students from the University of Michigan in fall 1956 to live for 16 weeks in an off-campus fraternity housing. None of the students had known anyone of the other students before the start of the experiment. After every week, every student had to rank all other students of the observed group from 1 (= best friend) to 16. Newcomb himself analyzed his data primarily with statistical methods showing that physical proximity, reciprocity, similarity, and complementarity were the main reasons to form friendship. The first three of these points became generally accepted theories for tie formation in the subsequent decades and we will describe their meaning in chapter 2.

Many scientists analyzed the networks that were created in Newcomb's (1961) study. They were particularly interested in the forming of the network structure (e.g. sub-groups). Most of the studies that re-analyzed Newcomb's fraternity data, came to the conclusion that the acquaintance forming process was more or less finished after four or five weeks and that not so much changed after this point. Nevertheless, the dynamic analysis of the networks and in particular the original study of Newcomb (1961) describing the underlying impulses of the acquaintance process were not possible without analyzing not just a single social network but a collection of networks over time.

DRAFT

1.2 Dynamic Network Analysis as answer

DNA is concerned with *who* is connected to *whom* (as in traditional SNA) and the strength, direction, and type of connection. In addition, DNA moves beyond the social to simultaneously examine *who* is in *what* groups, has *what* capabilities or expertise, is engaged in *what* activities, and holds *what* beliefs. DNA simultaneously puts these networks in a geo-temporal context that defines *who* was *where and when*. These diverse entities define a meta-network of connections that link *who*, *what*, *where*, *when*, *how*, and why (Carley, 2004a). In this book, we introduce the exciting field of DNA to the uninitiated and provide additional information on current research to the initiated. It is our broad and lofty hope that you will soon find yourself immersed in DNA and thus able to reap the powerful, insightful and critical analysis that only DNA makes possible.

Let us imagine you are the president of your own company of 45 professionals who work in the areas of software development, marketing, research, and distribution. Things were going well for years, but now profits are slipping. You begin to wonder if you are maximizing the resources you have at your disposal. Do you truly have the most talented people in the most important roles critical to your business? Is it time to look at how the skills in your company have changed? What if years go by and everybody has the same knowledge they did when they started. Chances are your knowledge base would become outdated and obsolete. You need to make sure, that does not happen so you plan to periodically monitor your employee's talent and knowledge base.

Do you need a tool to figure out if your company is set up in the best way possible but are not sure how to do it? So what do you do? Do you just take an educated guess about how your company should be arranged to maximize employee's talents, skills and resources? Wouldn't it be nice if there was a tool that could make a model of your company and based on the linkages of education, aspirations, responsibilities, access to resources, business skills, tell what your company really looks like? For instance, wouldn't you like to know who would constitute the weakest link in your company or the most underused employee from a standpoint of their knowledge? Who are the emerging leaders? What employees have the most knowledge and where are they located? Can they access the right resources for the tasks they are given?

Perhaps you want to study how your company might likely perform based on the removal of "John" since John is moving to an out of town job. Who might take his place? Does he have access to some resources that nobody else has access too? Was he a silo of special information? What if John performed critical functions that officially belonged to someone else? How would you know that? Along those lines, what are the informal channels that exist at your company? Does your catering manager actually know the whereabouts of your key personnel better than the respective executive assistant does?

We can't answer any of the aforementioned questions by simply making educated guesses about who really does what in a haphazard fashion. We need a scientific method to go about a true analysis of the company. We need to create a model that takes into account all the critical entities and how they are truly connected. We would like our tool to provide scenario-based analysis

DRAFT

as well. Moreover, we need it to consider incomplete or outdated information. Could such a tool allow us to even predict how your company might grow in the near future given the removal or addition of any key employee? The answer is DNA.

Everyone that has ever heard of a terror network or complex organizational structure has an intuitive idea about how such a network might be displayed and, hence, analyzed. Such a person might logically envision that any such terror network might have a leader and a group of underlings to carry out certain terror-oriented tasks. They may further conclude that such a cell could be plotted out on paper by denoting actors with dots and drawing lines between them symbolizing connections. Likewise, a complex organization might have a hierarchical structure with a president and board of directors sitting at the top. However, those who are not trained in the science of DNA will not realize what can be fully gleaned from network analysis when one takes into account the cross-disciplinary approach of computational mathematics and other social-science disciplines.

In such a science, complex factors are considered when conducting network analysis. For instance, much like the *Special Theory of Relativity* changed the way we think of *space* and *time* to something called *space-time*, we have to take a far deeper analytical approach to what we mean when we say network analysis. After all, networks are not like molecules—they can learn. Yes, that is right. We already went into how networks can be comprised of nearly anything. One thing you are well familiar with is that networks, the ones we are most often interested in analyzing, are made up of people and those people have a tendency to learn and forget, grow and decline. People also tend to react to certain events in different ways, which could easily change a network model. So let us be clear on this point: networks don't exist in a vacuum—they evolve (Bonacich, 2001). Networks don't suffer damage without responding in some way—be it growth or the emergence of new leaders and increased activity. Networks don't stay the same either—they change constantly. What you analyzed today is altered by the time it is read and considered by another. The change can be dramatic or small, but any change can be critical. One's assessment of any part of the network can be skewed if the information, on which your assumptions are based, proves to be false. Along similar lines, the information you have on a network might only be the tip of the iceberg. When you consider all of these quandaries, a more robust science is needed to carry out effective network analysis.

Below is a list of issues that can be tackled with DNA (Aldrich and Herker, 1977; Wasserman, 1980; Watts, 1999; Carley, 2001; **?**, 2003b; Carley et al.,

DRAFT

2003; Carley, 2004b; Carley et al., 2004):

- Developing algorithms to track groups in networks over time.
- Developing and testing theory of network change, evolution, adaptation, decay, etc.
- Developing control processes and statistically valid measurements for networks over time.
- Examining networks as probabilistic time-variant phenomena.
- Forecasting change in existing networks.
- Identifying trails through time given a sequence of networks.
- Identifying changes in node criticality, given a sequence of networks and anything else related to multi-mode multi-link multi-time period networks.
- Developing metrics and statistics to assess and identify change within and across networks.
- Examining the robustness of network metrics under various types of missing data.
- Empirical studies of multi-mode multi-link multi-time period networks.
- Developing and validating formal models of network generation and evolution.
- Developing and validating simulations to study network change, evolution, adaptation, decay, etc.
- Developing tools to extract or locate networks from various data sources such as texts.
- Developing techniques to visualize network change overall or at node level; or the representation of a single entity, or group level, which contains multiple entities.

DRAFT

In this book we will discuss some of these issues and provide you with the basic DNA knowledge and skills to perform your own network projects.

1.2.1 Who can use DNA?

DNA can be used by any type of analyst interested in state-of-the-art network analysis for a variety of reasons. This includes university researchers and analysts employed by various profit and non-profit corporations, military units, and government units. The ubiquity of networks means nearly any organizational analyst can use DNA to solve his or her own unique network problem

(Carley, 2006) no matter what field the analyst is involved in. Like statistics, DNA is a general purpose analytic tool that helps the analyst understand, assess, and predict social behavior. To understand what networks are best for DNA, we need to learn how DNA is applied and what process is involved in conducting DNA analysis. There are two aspects of DNA. The first is the statistical analysis of DNA data. The second is the use of simulation to determine how a dynamic network will evolve over time.

DNA networks vary from traditional social networks in that they are larger, dynamic, multi-mode, multiplex networks and may contain varying levels of uncertainty. Moreover, DNA statistical tools are generally optimized for large-scale networks and simultaneously admit the analysis of multiple networks in which there are multiple types of entities and multiple types of links (multiplex data). In contrast, SNA statistical tools focus on single or at most two node classes (two mode data) and facilitate the analysis of only one type of link at a time (Freeman, 2000).

DNA statistical tools tend to provide more measures to the user because they have measures that use data drawn from multiple networks simultaneously (Carley, 2003b; Breiger et al.). From a computer simulation perspective, entities in DNA are like atoms in quantum theory because the entities can be treated as probabilistic. Entities in a traditional SNA model are static, whereas entities in a DNA model have the ability to learn. In a DNA model the properties can change over time, and the entities can adapt (Breiger et al.). For example, a company's employees can learn new skills and increase their value to the network (Watts and Strogatz, 1998).

DNA allows us to analyze the interplay between various different types of *who*, *what*, *where*, *when*, and *how*–which are the entities that can pretty much include just about anything in the physical universe (Carley and Lee, 1998). DNA adds the critical element of a network's evolution and considers the circumstances under which change is likely to occur and how it applies to the entities that compose it (Carley, 2004a). It is with these entities that our analysis must begin.

1.2.2 More than "who you know"

DNA integrates different fields which have been developed in the past. Figure 1.2 gives you an overview of the genealogical tree of DNA. In short, DNA is rooted in graph theory and is combined with methods and theories of Anthropology, Sociology, and Organization Theory on the one side. The other side of DNA's family tree focuses rather on single links and the attributes of

DRAFT

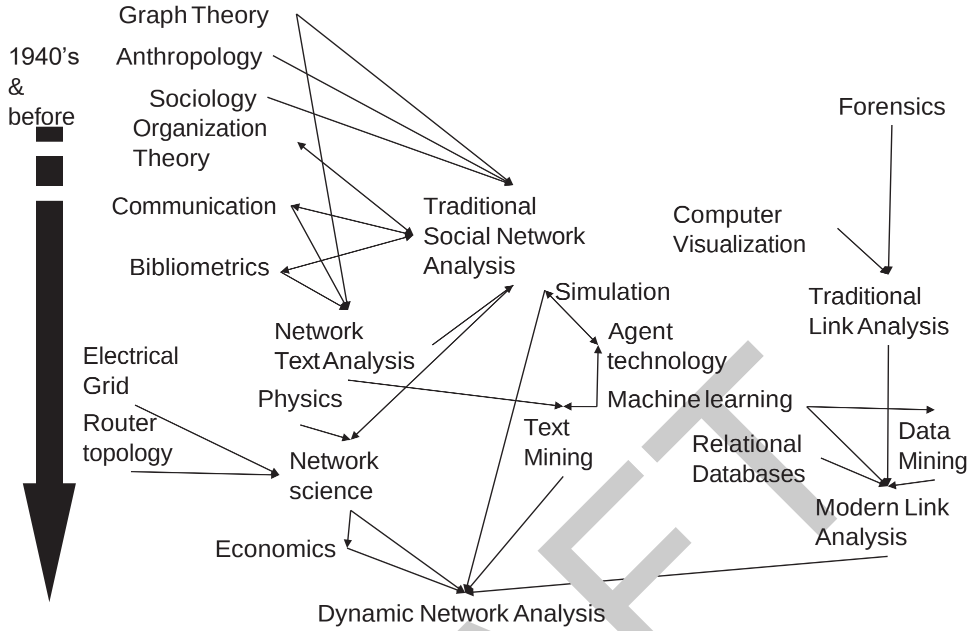


Figure 1.2: The genealogical tree of dynamic network analysis

DRAFT

agents connected through these links than on the structure of the network. In recent years, physicists working with network models have been creating a field of Network Science which is—when looking at the methods they are using—very similar to other areas enumerated in Figure 1.2. Nowadays, a lot of different research areas describe different aspects of network analysis. DNA includes the best of these worlds.

An entity in DNA is essentially the *who*, *what*, *where*, *how* and *why*. It is something that is being studied, and it exists in relation to something else. In addition, as we mentioned, networks can be made up of practically anything: you and your friends constitute a social network and in it as such represent entities in that network. For example, the millions of living organisms and food sources in an ecosystem would be considered network entities. Furthermore, a company might set up a network of computers and, thus, computers are now entities in that network as well. An isolated group of terrorists might constitute a cellular network since each terrorist is an entity of some sort. In addition, orbiting satellites can also be networked. Our solar system is a vast network of billions of stars of varying sizes and shapes. As a whole, these stars comprise the Milky Way galaxy. The Milky Way galaxy is one of many galaxies comprising the infinitely vast network of galaxies that constitute the

Universe. Are you beginning to notice a pattern? Have we made it clear that just about anything you think of can be considered a network on some abstract level and, as such, anything that makes up a network is an entity of some kind which interacts with other entities?

We will get into more details about the sorts of entities a DNA scientist typically deals with soon enough. For now, we want to consider another element that is very integral to an entity. This element is time. Time does not stand still and over any given period, things, that are entities, are likely to change. This is similar to how Albert Einstein made the connection between space and time. It is clear that networks occupy space, and it would only make sense to see that time is integral to the space in which the network exists. A network that exists this year can be dramatically different than the same network represented next year. DNA takes this change into account. Actually, the analysis of change is the main challenge of DNA.

Entities and their relations are always changing, and this makes networks dynamic. DNA is plowing the path for the study of this dynamic activity which was previously inaccessible in the traditional disciplines of link analysis where change was not likely to be a key factor. DNA looks at networks not merely as a bunch of people connected to other people, but people that exist in time and can be different, and often are, from one time period to the next. Some people, after all, learn new knowledge and forgot knowledge just the same. We are looking at networks in terms of their interconnectedness to other entity networks and how change occurs as time marches on (Carley, 2001). This is where DNA proves its mettle. With DNA we can take a snapshot of a network in time and with some skill in analysis, stay one step ahead of the curve.

Let us now consider DNA on a more practical level, in a manner that might help explain situations you have probably encountered many times. The importance of networking is something we have all encountered before in one form or another–be it in our personal or professional lives. In such everyday experiences, we might say that networking is the art of making meaningful connections. We have all heard the expression: *It is not what you know but who you know.* Let us consider this common morsel of wisdom from the perspective of DNA.

Countless variations of the phrase *It is not what you know but who you know* ring true across many boundaries from the cynically hardened skeptics to the most incorrigible optimists. Moreover, such a turn of phrase is often ascribed as the key to both personal and professional success. However, what is this phrase really hinting at? We know it describes a network but what

DRAFT

exactly about the network? The laws of sociological nature? The laws of social dynamics? How to get ahead? The art of networking? A certain part, or aspect, of what it means to be an important component of a network? All of the above? None of the above? Some of the above?

The truth is that the simple phrase, *It is not what you know, but who you know* describes merely one single facet of any social network, and the dynamic network analyst would find this turn of phrase incredibly simplistic. In fact, have you ever considered it might totally be wrong? Can you think of examples where the opposite might be true? How about a network model of a Ph.D. program where an advanced degree is conferred by accumulating and presenting research than defending such research until the thesis is accepted? In this network, could it not be argued that *what you know* is more important than *whom you know*? Perhaps. Nevertheless, back to that hackneyed phrase: *It is not what you know, but who you know*. What is to be made of it from the stand point of DNA?

1.3 Caesar, Brutus, and co.

To understand DNA more fully, we will apply the tool to *The Tragedy of Julius Caesar* as crafted by William Shakespeare. Why Julius Caesar? In short, we have chosen *The Tragedy of Julius Caesar* because chances are it is a literary work many of us have encountered at one time or another in our educational backgrounds, whether it be from high school or at the post-secondary level. Moreover, it is about a simple usurpation of power, an assassination, and betrayal. Meanwhile there are conflicting values, a plot, and a network of people who made decisions. There is a lot of network complexity in that play. Therefore, in our opinion, *The Tragedy of Julius Caesar* is a highly useful network from that standpoint: neither too big nor small. It is just right to show the power of DNA properly with a model that you may likely know already. It is especially useful because it is made up of a fairly complex arrangement of characters, allegiances, and resources. With the use of DNA and a time machine, we might even be able to suggest to Julius Caesar how he could have prevented his own demise.

You need not re-read the play to understand the examples we will go through in our application of DNA. However, a familiarity with The Tragedy of Julius Caesar might help you get more out of this book. We suggest *Sparknotes.com* for a short but concise summary of the characters, events and plot (simply search “Julius Caesar” on the SparkNote’s site search function). You can also get a copy of Cliff’s Notes from your local bookstore. Better yet, purchase a

DRAFT

copy of the play, dust off the old one in your book collection, and do something novel, like read the play over again. It shouldn't take you more than a couple hours. You might even enjoy it.

It is our hope that when presented with the proper DNA model, even Brutus would have seen that the fault surely did not "lie in the stars" as Cassius reminded him in course of events. Rather, the fault lies in the failure to analyze a complex multi-modal network of Roman politicians, plebeians, military leaders, poets, family member, citizens, soothsayers, battles, skills, allegiances, knowledge, rhetoric and what have you–that is where true fault resides.

So we begin a journey, in hindsight nonetheless, to analyze *The Tragedy of Julius Caesar* by William Shakespeare, and offer our own analytical recommendations and insights surrounding Caesar's assassination by putting the power of DNA to work on the network of Julius Caesar as extrapolated from the legendary play *The Tragedy of Julius Caesar*.

1.3.1 The plot of the tragedy

Let us begin to consider DNA in the context of our specially created example solely for the illustrative purposes of this book. *The Tragedy of Julius Caesar* by William Shakespeare represents the assassination against the Roman dictator Julius Caesar, including the aftermath. It is based on true events from Roman history. Although the title of the play is *The Tragedy of Julius Caesar*, Julius Caesar is not the central character in the plot, as you will soon learn. In fact, Julius Caesar only appears in three scenes and he dies at the beginning of the third act. The protagonist of the play is actually Marcus Brutus. The plot focuses on his internal struggle with the conflicting demands of honor, loyalty, and companionship.

The play begins with two tribunes named Flavius and Marullus who discover a large crowd of Roman citizens roving the streets. The pedestrians are celebrating Julius Caesar's victory over the Roman general Pompey, his archrival, during a battle. The tribunes scold the citizenry for abandoning their duties and instruct them to remove the decorations from Caesar's statues. Caesar enters with his associates, including the military and political figures Brutus, Cassius, and Antony. Famously, a Soothsayer calls out "beware the Ides of March," but Caesar ignores him and continues with his victory celebration.

Later, Cassius and Brutus, who are friends of Caesar and each other, begin to confer. Cassius tells Brutus that he seemed withdrawn recently. Brutus responds by saying that he is full of self-doubt. Cassius replies by voicing his wishes that Brutus could see himself as others see him. Cassius goes on to

DRAFT

explain that if Brutus had more confidence in himself he would realize how honored and respected he is. Therefore, in turn, he would feel more secure in his rightful place. Brutus states that he worries the people want Caesar to become king, which would overturn the republic and convert it into an authoritative regime. Cassius agrees with Brutus, and they point out that Caesar is already considered to be a god-like figure that people idolize. In an effort to empower Caesar, Cassius reminds Brutus that Caesar is only a man, and he is not superior to Brutus or Cassius. To back up his claims, Cassius recounts incidents of Caesar's physical weakness and expresses his shock that this fallible man has become so powerful. He blames his and Brutus's lack of conviction for allowing Caesar's rise to power. Brutus considers Cassius's commentary as Caesar returns. Upon seeing Cassius, Caesar lets Antony know about his suspicion and distrust for Cassius.

After Caesar departs, a politician named Casca tells Brutus and Cassius that, during the celebration, Antony offered the crown to Caesar three times. Each time the crown was offered the people cheered, but Caesar refused it every time. Casca reports that right afterwards, Caesar fell to the ground and had some kind of seizure in front of the crowd. While some would consider this to be a sign of weakness the plebeians were unaffected by it, and continued to show their devotion to him. Later, Brutus considers Casca's observations that suggest Caesar's poor qualifications to rule. Meanwhile, Cassius brainstorms a plan to involve Brutus in a conspiracy against Caesar.

That evening, Rome experiences destructive weather and a variety of bad omens and forewarnings. Brutus finds letters in his house that are supposedly written by Roman citizens who are worried that Caesar has become too dominant and controlling. In actuality, the letters have been fabricated and planted by Cassius. Cassius does this because he wants Brutus to believe that the public is dissatisfied with Caesar. He knows that Brutus is deeply affected by the republic's reaction, and, therefore, after reading the letters he knows that he will likely become more supportive of Cassius's plot to remove Caesar from power. Brutus fears that the populace would lose its voice in a dictator-led empire. When Cassius arrives at Brutus's home with his conspirators, Brutus is already influenced by the letters, and he takes control of the meeting. The men unanimously agree to lure Caesar from his house and murder him. In addition, Cassius wants to kill Antony too. His logic is that Antony will ruin their plans. Brutus refuses to murder Antony since he fears that too many deaths in their plan will appear too bloody and dishonorable. After they all agree to spare Antony, the conspirators depart. Brutus's wife, Portia observes that Brutus appears distracted and ill at ease. She begs him to confide in her, but he ignores her.

DRAFT

As Caesar continues to prepare to go to the Senate, his wife, Calpurnia, begs him not to go as well. In an effort to persuade him, she describes recent nightmares she has had. In the nightmares she envisions a statue of Caesar covered with blood and smiling men bathing their hands in the blood. Caesar refuses to react to fear and insists on going about his normal routine. Eventually, Calpurnia convinces him to stay home. He agrees to stay home only as a favor to her, and is careful to point out that his decision is not based on fear. However, soon his plans change when Decius, one of the conspirators, arrives. He assures Caesar that Calpurnia has misinterpreted her dreams, as well as the recent omens. Caesar heads toward the Senate with the conspirators. As Caesar proceeds through the streets toward the Senate, the Soothsayer once again tries to warn him. However, his attempt to get his attention is unsuccessful. In another attempt to warn Caesar, Artemidorus, a citizen, hands him a letter to advise him about the conspirators, but Caesar refuses to read it. While at the Senate, the conspirators speak to Caesar. As they are huddled around him, they take turns stabbing him to death. When Caesar sees his close friend Brutus among his murderers, he stops resisting the attack and dies.

Calpurnia's prediction comes true when the murderers bathe their hands and swords in Caesar's blood. Antony returns, after having been led away on a false pretext, and vows his allegiance to Brutus. Later, however, he weeps over Caesar's body. He shakes hands with the conspirators, thus making them all appear guilty while trying to make a gesture of conciliation. When Antony asks for an explanation as to why they killed Caesar, Brutus replies that he will explain their reason at the funeral. Antony asks to be allowed to speak at the funeral, and Brutus grants his permission. Cassius, however, remains leery of Antony. After the conspirators depart, and Antony is alone, he asserts that Caesar's death must be avenged.

Later, Brutus and Cassius go to speak at a public forum. Cassius exits to speak to another section of the crowd. Brutus explains to the crowd that although he admired Caesar, his ambition put Roman liberty at risk. The speech pacifies the crowd. Brutus turns the pulpit over to Antony when Antony appears with Caesar's body. Antony's speech begins with praise for Brutus, but then becomes increasingly sarcastic. He questions the statements that Brutus made in his speech that Caesar acted only out of ambition. Antony calls attention to the wealth and glory that Caesar brought to Rome. However, with all the success that Caesar had, he rejected the crown three times. Antony points out that Caesar was clearly not solely interested in the power to rule. Antony takes out Caesar's will with the intention of reading it, but then he stops himself from reading it since he decides that it will cause unnecessary

DRAFT

distress to the people. Nevertheless, the crowd pleads for him to read the will. He leaves the pulpit to stand next to Caesar's body. He describes Caesar's abhorrent death and presents Caesar's wounded body to the crowd. Afterwards he reads Caesar's will, which states that a sum of money will be given to every citizen and orders that his private gardens shall be made open to the public. The fact that such a generous man was horribly murdered enrages the crowd, and the crowd begins to call Brutus and Cassius traitors. The masses begin their plan to eject them from the city.

In the meantime, Caesar's adopted son and appointed successor, Octavius, arrives in Rome and forms a pact with Antony and Lepidus. They prepare to fight Cassius and Brutus, who have been driven into exile and are raising armies outside of the city. Brutus and Cassius have a heated argument regarding money and honor, but they ultimately decide to settle their disagreements. Brutus reveals that he is grieving the death of Portia, who committed suicide. The two continue to prepare for battle with Antony and Octavius. The Ghost of Caesar appears to Brutus that night. It announces that Brutus will meet him again on the battlefield.

As Octavius and Antony march their army toward Brutus and Cassius, Antony instructs Octavius where to attack, but Octavius stubbornly replies that he will make his own orders. He is eager to assert his authority as the heir of Caesar and the next ruler of Rome. The rivaling generals meet on the battlefield and exchange harsh words to each other before beginning to fight.

Cassius begins to notice that his own men are retreating and he hears that Brutus's men also are not performing effectively. Cassius sends one of his men, Pindarus, to check on the situation. From afar, Pindarus sees one of their leaders, Cassius's best friend, Titinius, being surrounded by applauding troops and infers that he has been seized. Cassius becomes distraught and orders Pindarus to kill him with his own sword. He dies after proclaiming that Caesar is avenged. Soon after, Titinius arrives, and it is revealed that the men who were encircling him were actually on his team, and they were celebrating the victory over the opponents. When Titinius sees Cassius's corpse he begins to mourn the death of his friend. He is so distraught that he kills himself.

When Brutus learns of the deaths of Cassius and Titinius he is also upset, and he prepares to take on the Romans once again. When his army loses the battle, Brutus asks one of his comrades to hold his sword while he impales himself on it. As he is dying, he proclaims that Caesar can rest satisfied. When Antony speaks over Brutus's body, he calls him the noblest Roman of all. He points out that while the other conspirators acted out of envy and ambition, Brutus genuinely believed that he acted for the benefit of Rome.

DRAFT

Octavius orders that Brutus be buried in an honorable way. Afterwards, the men leave to celebrate their victory.

This was the story of the Tragedy of Julius Caesar by William Shakespeare. Now that we got that out of the way, it is time to get down to some DNA. After all, now that we know the story, now we need to know the nodes (the whos) that will make up our meta-network. And, without further adieu, we are ready to talk about the basic building blocks of network analysis.

1.3.2 Saving Julius Caesar?

Nearly everything is a network. The universe is expanding. Your knowledge is growing or languishing. People move on to different roles. One day you're a son, the next day you are a parent. Like string theory and quantum mechanics, everything in our vast interconnected universe is, on some level, constantly on the move and this is what you will come to see in Julius Caesar. The time element makes depicting network models especially tricky because no sooner than you construct a network, it has changed. Since this applies to Julius Caesar, we will explore several techniques that will help you properly account for time in your own network model.

Using DNA, our aim is to discover what Julius Caesar could not discern for himself; how he was vulnerable in his own empire by failing to understand the complex multi-modal evolving network around him. In doing so, we will introduce and explore the power of DNA. We aim to show how this is done based on our knowledge of the Julius Caesar network as presented by Shakespeare. A DNA analyst could have made certain recommendations, based on rock-solid mathematical computations, to Caesar, which might have seen him carrying on his rule as emperor of Rome and conquering the rest of known world, as he probably would have liked to have done.

Although a skeptic might conjecture that he too would have ignored our insights, much as he did the dire warnings of the Soothsayer, the nightmares of his wife Calpurnia, and the advice of Artemidorus shortly before his ill-fated trip to the Roman Senate. But, that only underscores a human volatility that can in part affect the impact of a well put together network model: is the person who is looking at the model shrewd enough to see what it really is?

We know one thing: if we presented Julius Caesar our findings, based on the authority of the cutting-edge computational mathematics of DNA, Decius would have had a much harder time convincing Julius Caesar that his wife's dream was merely misinterpreted and that he *should* attend the Senate meeting that day, where he would promptly be stabbed by his closest friends. Our

DRAFT

findings would have given him much pause. Still, Caesar would have to act upon them by making some kind of policy decision. He seemed to “go at it alone,” and it cost him his life.

Nonetheless, rooted in the knowledge of DNA, a better policy for Julius Caesar could have been crafted to avert his impending doom. We can take solace in our lesson, however, that grounded with the results of better analytical methods we might construct policies that would prevent a network from doing the same again be it for nefarious purposes or altruistic. Along those lines, we should add that our purpose is not necessarily to show how DNA can buttress the continuation of a tyrant, as Cassius might have argued, but just demonstrate how useful it can be when applied with foresight and skill. Brutus could have also used his own DNA model, perhaps, to see the likely outcome of killing Caesar. He might not have needed an ill dream to tell him that Rome would be divided in two. He might have only needed his DNA model.

Therefore, before we begin to build our network model of the Julius Caesar world, we first need to explain to Caesar what a network is, what its components are, what the best ways of analyzing them are, and what challenges are faced in analyzing complex multi-modal networks over periods. We begin with the basics for those Roman citizens of network analysis lacking in the rudiments as we are sure Caesar would have been in the same class as the soothsayers and cobblers in that regards.

DRAFT

1.4 Problem set

1. Remember the study by Allen and Fustfeld described in this chapter—why is the desk-to-desk distance in a company important for collaboration?

2. What is the main difference between social network analysis and dynamic network analysis regarding to the entities of the analysis?

3. Imagine the network consisting of all employees in your company or all students of your university as well as their connections based on sending and receiving e-mails. What are the questions which you want to have answered with such network data?

4. Thinking of accomplishing tasks, why do you think is it important to know which person in an organization has which knowledge?

5. Let us assume Julius Caesar had the support of a lot of smart network analysts (including yourself after finishing this book). Why do you think that Caesar would have listened to his analysts more that to the soothsayers?
6. What are the differences between link analysis and network analysis?
7. Lievrouw et al. (1987) created networks from words occurring in texts. What are the links in these networks?
8. Do you have a better network than your best friend? What are the problems with this question?
9. Imagine a data table with people as rows and socio-economic attributes for the people as columns. Why are these data not suitable for network analysis?
10. Johnson et al. (2001) analyzed food chain networks. What can be a possible research question using these data?
11. *Imagine a data table with people as rows and socio-economic attributes for the people as columns. Why are these data suitable for network analysis?
12. *What are the advantages and disadvantages of networks extracted from social media?
13. *Read Crane's invisible college study and discuss whether the results of this work are still valid today, in particular, in your area of research.
14. *Download the Julius Caesar network data from the book's website. Install your favorite networks analysis tool and load the Julius Caesar networks. That's all for now, you will see how to analyze this data in the next chapters.
15. *Why do you think is your social network important for getting a good job. Also, collect some information connected to "Getting a Job" by Marc Granovetter.
16. *Reading this book, you are probably a PhD student or an analyst that spent lots of time in previous years with learning to know a lot. Do you think "it is not what you know but who you know" is correct? Do you think that your social network is the result of your personal characteristics (education, personality) or are your characteristics the result of your network structure?

DRAFT

17. **Find the Newcomb fraternity data on the web and run some simple statistical analysis to show which students are in vogue and how their popularity changes over time.

18. **Search online for an article that covers your area of research and that uses network analysis as a method. What is accomplished in this article? What are your concerns?

19. **What analysis are you planning to accomplish by using network analysis? What are your hypotheses? What data are necessary to perform network analysis?

20. **List ten people of your closest friends, co-workers, or fellow students. Collect the following information about these people (you can ask them if you do not know this information): A) Who knows whom. B) Who is a good fried of whom. C) What are the organizational affiliations of these people (e.g., universities, companies). D) What are there areas of expertise (knowledge). What do you learn from the data? How are the connections of data collected for A, B, C, and D? Based on these data, is somebody important in your network? Why?

DRAFT

Chapter 2

Analyzing Social Networks

DRAFT

Social networks are networks consisting of human beings. The people in these networks are often called *Agents*, nodes, or actors. These terms are used interchangeably. Connections between these *Agents* are called edges (or links). These two sets, nodes and edges, are sufficient to describe the essence of social networks. Therefore, before we can start to collect and analyze the network of Julius Caesar, we have to answer the two fundamental questions of social network analysis: What are the entities of our networks? And how are these entities connected with each other?

2.1 Units of interest

2.1.1 Entities: Friends, romans, and countrymen

In the following list, we have the characters that make up our Julius Caesar Network, which is based on William Shakespeares *The Tragedy of Julius Caesar*. All of the characters on this list constitute an entity, which for the purposes of DNA we say is a *who*. After all, they are people, even though it is in a fictional sense. Other entity classes, which you will see later in this book, allow us to put certain entities into other different containers, which, perhaps you have guessed by now, correspond to the *who, what, when, where* and *why* model. There are even more components as well, but we will explore those more deeply when we learn about entity classes. For now, let us visit our *whos* as they relate to the Julius Caesar Network we are going to build in the next chapter. Here are the *whos*!

By reading the names of characters in Table 2.1 you can imagine that the

Antony, (Marcus Antonius)
Artemidorus, a teacher of rhetoric
Brutus, (Marcus Brutus)
Caesar, (Julius Caesar)
Calpurnia, wife to Caesar
Casca, a conspirator against Caesar
Cassius, a conspirator against Caesar
Cicero, Senator
Cinna the Poet
Cinna, a conspirator against Caesar
Citizens Publius, Senator
Claudius, servant to Brutus
Clitus, servant to Brutus
Dardanius, servant to Brutus
Decius Brutus, a conspirator
Flavius, a tribune
Lepidus, (Marcus Antonius Lepidus)
Ligarius, a conspirator against Caesar
Lucilius, friend to Brutus
Lucius, servant to Brutus
Marullus, a tribune
Messala, friend to Brutus
Metellus Cimber, a conspirator
Octavius, (Octavius Caesar)
Pindarus, servant to Cassius
Poet
Popilius, (Popilius Lena)
Portia, wife to Brutus
Publius, Senator
Soothsayer
Strato, servant to Brutus
Titinius, friend to Brutus
Trebonius, a conspirator
Varro, servant to Brutus
Volumnius, friend to Brutus
Young Cato, friend to Brutus

Table 2.1: Entities (cast) of characters in Julius Caesar (*whos*)

DRAFT

decision of *who* is in the network and *who* is not, is not always trivial. It is obvious that the main characters of the play such as Caesar, Brutus, and Antony should represent nodes of our network. But, how should the other characters who just play supporting actors be handled? And what about those who do not even have a name? For example, citizens sometimes speak as a group and sometimes a single citizen is labeled "First Citizen" or "Second Citizen."

First, lets discuss the poets. In the entity list of Table 2.1 you will find two poets, "Cinna the Poet" and the "Poet." The first one has a name, which makes him a *specific* entity. The second poet is a *generic* entity. When it comes to generic entities we always have to be careful to be sure that different references to these entities really discuss the same *Agent*.

We decided to add the poet who shows up in the fourth act to tell Brutus and Cassius that they should stop fighting with each other. We know that this poet is different from Cinna, the poet who gets killed accidently in after the assassination of Caesar. A couple of "Servants" are not part of our node set because we perceive them to be less important. With some exceptions, the servants of Brutus and Cassius have names which persuaded us to add them as *Agents* to our selection.

Cinna, the poet is also a good example for another challenge. We mentioned earlier that he got killed by the citizens because they confounded him with another guy called Cinna who was one of the conspirators against Caesar. What happened to the citizens of Rome can also happen to us when we are collecting data for our Social Network Analysis (SNA). Unifying different people to one entity of our analysis can easily create interesting artifacts—or, in other words, it can destroy your whole analysis!

Finally, the most critical of our decision is certainly the node "Citizens," which is a group node that represents a couple of anonymous people. We can use the same arguments discussed for generic nodes to ignore these nodes in our networks. For your own networks you should not mix up individuals and groups in one node class unless you have a really good argument for that. In our case, we think we have a good reason to lump them together. The Citizens are of key importance when it comes to the succession of Caesar and, less gloriously, they kill Cinna the Poet. In addition, we also want to refer to these incidents in the networks. But, we must be aware of the implications for calculating measures for our networks. We will discuss these implications in the later chapters of this book.

DRAFT

2.1.2 Relations: To love and to hate

Now that we have the nodes, the next fundamental decision is to determine how to connect the nodes with each other. In creating our social network of *The Tragedy of Julius Caesar*, we can say that the characters in the play constitute a social network based on their interactions. It will be a *who* by *who* network. Since the elements of our network are people, this is often called a social network. A social network should be something familiar to anyone. Whomever you regularly talk with can be a social network. It can be your friends, family, the people you work with, or any combination of them. This social network will be our first network, and it will tell us who is connected to whom. In the case of *The Tragedy of Julius Caesar* the *who* elements are the persons of the play as we described in the previous sub-section.

Now that we have all of the characters recorded, we need to figure out *who* is connected to *whom*. The question to answer is what do we consider as being connected? In our case, we are going to consider that they are connected with each other if they appear in the same scene and interact with each other. Let us start *coding* our network. Actually, we need to create five networks instead of just one network—every act in the play of Julius Caesar gets a separate network. The multiple networks will allow us to examine change over time later on. But, let us start with the first act and discuss over time issues later

		1	2	3	4	5	6	7	8	9	10	11	12	13	14
Antony	1	.	.	x	.	x	x	.	.	.	.	.	.	.	.
Brutus	2	.	.	x	.	x	x	.	.	.	.	.	.	.	.
Caesar	3	x	x	.	x	x	x	.	.	.	.	.	.	.	x
Calpurnia	4	.	.	x	.	.	.	.	.	.	.	.	.	.	.
Casca	5	x	x	x	.	.	x	x	.	.	.	.	.	.	.
Cassius	6	x	x	x	.	x	.	.	x	.	.	x	.	x	.
Cicero	7	.	.	.	.	x	.	.	.	.	.	.	.	.	.
Cinna	8	.	.	.	.	.	x	.	.	.	.	.	.	.	.
Citizens	9	.	.	.	.	.	.	.	.	.	x	.	x	.	.
Flavius	10	.	.	.	.	.	.	.	.	x	.	.	x	.	.
Lucilius	11	.	.	.	.	.	x	.	.	.	.	.	.	.	.
Marullus	12	.	.	.	.	.	.	.	.	x	x	.	.	.	.
Octavius	13	.	.	.	.	.	x	.	.	.	.	.	.	.	.
Soothsayer	14	.	.	x	.	.	.	.	.	.	.	.	.	.	.

Table 2.2: Social network matrix of the first act of Julius Caesar

DRAFT

in this chapter.

In Table 2.2 you can see a first representation of a network. You can see that the relations between the entities of a network are stored in a matrix. In the left column of the matrix as well as in top row, the *Agents* of our network are enumerated. An "x" is drawn in a cell of the matrix if there is a connection between two *Agents*, e.g. the Soothsayer warns Caesar about the *Ides of March*; therefore, you can find an "x" connecting Caesar with the Soothsayer. The network matrix is symmetric because you can find every connection twice in the network matrix. All other connections are set the same way, so that the matrix aggregates all connections of the three scenes of the first act of the play.

The decision making process that is involved in deciding which *Agent* is part of the network and which is not, is similar to the process of deciding what is an interaction and what is not. In the third scene of the first act, for example, the two conspirators Casca and Cassius are having a discussion when Cinna, another conspirator, enters the scene. Cinna has a short conversation with Cassius, but Casca is not involved in this conversation at all. After Cinna leaves the scene again, Casca and Cassius continue their interrupted conversation. So, if we build up a network based on occurring in the same scene we would establish a relationship between all three *Agents*, but if the network is based on actual interactions, we cannot set a connection between Cinna and Casca. For the networks of this book, we decided to look for interaction between the actors. Therefore, this scene leads to two links in our network, one between Casca and Cassius and a second between Cassius and Cinna.

This little example should show you that the decision whether a link is in a specific network or not can be tricky. In general, the definition of a link has to be made in every DNA project. When it comes to your own network projects, this is an important question to ask even if you are working with data collected by other people. In the next chapters, you will learn a lot about different measures to identify interesting nodes or groups of nodes or other patterns. These measures very much rely on your data. The networks of the same organization or other group of people, or even the networks of a play by Shakespeare, can look very different based on the definition of *who* is in the network and *who* is not, as well as the decision about which kind of connections to observe and which not to observe.

2.2 More definitions

This book is an introduction of network analysis, and it is targeted to students in all majors as well as non-academia people. Therefore, we avoid complicated definitions and equations in larger parts of the book, and we have put all of the mathematical details in the glossary part at the end of the book. Nevertheless, some basic definitions are necessary to be sure that we are all talking about the same things. First of all, when reading the first pages of this book, you have already seen that a *network* is defined by a set of *nodes* (e.g. *Agents*) and *links* connecting these nodes. A link which connects a node with itself is called a *self-loop*. Two nodes are neighbors if there is a link connecting this pair of nodes directly with each other. Two nodes are indirectly connected if there is a path through the network connecting a node with another node by intermediate nodes. For example, if *a* is connected to *b*, *b* is connected to *c*, and *c* is connected to *d*, then *a* and *d* are indirectly connected. The shortest indirect connection (using the smallest number if intermediate nodes) is called *shortest path*. The longest shortest path in a network is the *diameter* of the network. A group of nodes connected by direct or indirect connections is called a *component*. If you look at Figure 2.4 you can see that in act 1 and 4 our network consists of 2 components while in the acts 2, 3 and 5 all nodes are part of one single component.

The edges can be of different types. In our five Julius Caesar networks the edges are *unweighted*. This means that the importance of every edge is the same. Each has an “x” in the matrix and they all have lines with the same width in the network visualization. If it would not be the case that all the

DRAFT

edges in our networks are treated identically, the network would be weighted. We could, for example, say that the relation between Caesar and his wife Calpurnia is much more important than the relation between Caesar and the Soothsayer and, consequently, we are interested in coding this fact into the network model. We can do so, by putting different numbers into the network matrix, e.g. "5" into the matrix cells connecting Caesar and Calpurnia and "1" to his connection with the Soothsayer. If we write different edge weights into the network matrix, we call it a *weighted* network. To also represent the different line weights in the network visualization, we draw the lines with different widths.

In the context of edges we can introduce another definition. A network is called *undirected* if case A is connected to B and B is connected to A. It is *directed* if the connection is just in one direction. Why is this important? The Julius Caesar networks we constructed earlier in this chapter are undirected because our definition of a single connection is an observation while reading the book. If we had the ability to jump back in time and ask Caesar and his contemporaries with whom they interact with, our network data would look different. Why? Imagine an ancient scientist surveying Caesar about his social interactions. Probably this would turn out to be a long lasting interview because Caesar interacts with a lot of different people. There is, however, a pretty good chance that Caesar misses some people in answering this question; maybe he would not remember that he ever talked with the Soothsayer. On the other hand, a lot of people in ancient Rome who had contact with Caesar would recall him as one of the first names in their enumeration of connected people. To handle this asymmetric information about a single relation, we use directed edges. So, we are able to add a connection from the Soothsayer to Caesar but not vice versa.

DRAFT

We use the term *simple* networks to refer to networks which just consist of undirected and unweighted links, which have no self-loop, and which include all nodes in a single component. When considering the question about weighted or directed networks, you will realize that in social life almost no connection between people is undirected; even the most fundamental emotional connections, like love and hate, are somehow directed information. And of course, adding weights can help to describe these relations more in detail, e.g. how often do the people communicate, how close do they feel, or how long do they know each other. Nevertheless, in real world scientific and business SNA projects, the most networks are unweighted and undirected. These networks are easier to collect and to handle. A lot of measures ignore the direction or the weight of edges at all. In this book, you will find both variation; sometimes we will use networks with more characteristics. For the first measures we are going to introduce later in this chapter, just simple networks are used.

2.3 The structure of human connections

Before we start to analyze social networks, we want to give some attention to the question why people interact with each other at all and which network structures emerge as a result of social relations.

2.3.1 Networks on personal level

Think about the people that you met today or that you send emails to before you started to read these pages, why did you interact with them? Or why do Flavius and Marullus as well as Brutus and Cassius communicate with each other rather than all of the four as a group? During the last couple of decades scientists figured out that a vast part of the dynamics that drive the formation of social connections can be described with a few theories.

Reciprocity. The simplest answer to the question, "why does A communicate with B" is: "Maybe, B communicated with A the day before". Network analysts call the tendency of these bi-directional relations *reciprocity* (Katz and Powell, 1955). Following this theory, people interact with each other because there is communication history. There is a good chance that you send an e-mail to John because John had sent you an e-mail before, asking you a question or telling you some news. Cassius confides his doubts with Caesar to Brutus because he trusts him, probably because they have discussed other serious matters before. Consequently, when you observe networks over time, reciprocal connections will be a phenomenon that you will find very often. "Reciprocal service" is also one of the characteristics that describe *strong* ties (Granovetter, 1973) – we discuss weak and strong ties in some paragraphs more in detail. In opposite, very asymmetric relations, i.e. A sends a lot of information to B but B does not answer very often or maybe never, are a good indicator for hierarchies (e.g. A is an employee of B and has to report him about his project progress) or possession of valuable *Knowledge* or *Resources* (Wellman, 1988, p. 45).

Homophily. The second big theory of social connections is *homophily*. Homophily that people tend to connect with other people that are similar to themselves. Similar based on age, gender, education, or any other socio-demographic characteristic as well as similar behavior, goals, political positions, and religious *Beliefs*. Homophily is, therefore, what people mean when they use the saying "birds of a feather flock together". In their review arti-

DRAFT

cle about homophily McPherson et al. (2001) emphasized the importance for social connections when they stated:

> “Homophily limits peoples social worlds in a way that has powerful implications for the information they receive, the attitudes they form, and the interactions they experience.”

The fact that people create connections based on homophily has very important implications for the formation of groups – we will discuss this topic in the next section. For now, we can answer the initial question of this section about the relations in the Tragedy of Julius Caesar simply with “perhaps Flavius and Marullus as well as Brutus and Cassius interact because they are very similar to each other”.

Propinquity. A special form of homophily is similarity based on physical *Locations*. Festinger et al. (1950) figured out in their *Westgate West Study* that the friendship of students within a dorm is a function of the closeness of their rooms. Physical (geographical) closeness is very important impetus to forming relations – in other words: “Those close by, form a tie”. This is not just true for students but also for co-workers or neighbors in suburbs.

Transitivity. So far, we have just discussed the direct interactions between two people. The next theory about why people form relations with each other involves a third person. Transitivity describes the tendency that people form new relationships with friends of existing friends. In other words, if Jane and Betty are friends for many years and Jane also is good fried with Joe, than there is a good chance that Betty and Joe also form a positive relation with each other. The dynamics in these triadic relationships where introduced by Heider (1946) in his *Balance Theory* that describes the interaction of two people and their relationship to a third person (Cartwright and Harary, 1956) or any other form of entity (e.g. *Event*, idea, etc.). Look at Figure 2.1, triad number 1 shows a balanced triad, i.e. A and B like each other and they both like C. In opposite, triad 2 is unbalanced since A likes C but B doesnt. Why is this unbalanced? Imagine once again the two friends Betty and Jane and a slightly changed situation in which Betty likes Joe but Jane doesnt. What do you think happens in such a situation? Right, Betty and Jane start to argue about whether Joe is a nice guy or not. This results in an unbalanced triad. And the dynamics of this arguing between Betty and Jane are also quite predictable. Actually, there are two very likely outcomes. First, Betty convinces Jane that Joe is a nice guy. In this case the balanced triad number 1 would be the result. Second, Jane convinces Betty that Joe is not a nice guy. The result is triad number 3—which is again a balance

DRAFT

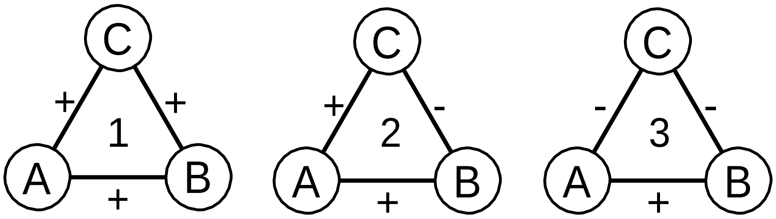


Figure 2.1: Triads of Heider's balance theory

triad as the double negative relation stabilizes the relation between Jane and Betty. [1] We can find another—yet more complex—example for transitivity in the Julius Caesar play. Return your mind to how the conspirators prepared their attempts to convince Brutus to support their plans. They sent him fake letters from residents of Rome talking about the concerns over Caesar and his leadership. When the conspirators showed up at Brutus house, Brutus closed a pretended balanced triad with the conspirators and the people of Rome!

2.3.2 Networks on group and society level

After we discussed how individual relationships are formed, let us continue these considerations at a larger level. If we *zoom out* and look first at network consisting of all relationships of a single person (e.g. of you), and in the next step at networks of groups (e.g. a group of friends) or even entire cities, we can find the following characteristics. But let us start with two theories that give better insight into arrangement of connections around a single *Agent*. In case we talk about a focal single *Agent* and his connections, we call the focal *Agent* the *ego*, the other nodes that are connected to ego we call *alters*, and the network with all its nodes and edges is referred to as *ego network* or *personal network*.

Weak/Strong Ties. People have a lot of relationships, some of which are subjectively more important than others. Brutus is a very good friend of Caesar (at least he thinks that he is) while Caesar does not feel so close to the other Senators. Mark Granovetter used the words *weak* and *strong* ties for these different relationships (Granovetter, 1973). He also offered a definition for classifying the strength of a connection:

> "The strength of a tie is a (probably linear) combination of the

[1] Of course, there are some more possible triads in case we do not just look at symmetric but also at unsymmetric relationships, but the basic ideas and dynamics stay the same.

DRAFT

> amount of time, the emotional intensity, the intimacy (mutual confiding), and the reciprocal services which characterize the tie." (Granovetter, 1973, p. 1361)

Based on this definition, every person has a small number of strong ties and a more or less large number of weak ties (see next paragraph). *Strong ties* are our best friends and parts of our core family. These connections are very important for our everyday life as well as for support in important situations. But, Granovetter's article (1973) was titled "The Strength of Weak Ties". He wanted to emphasize in particular these connections in the context of the diffusion of information. To give you an example why weak ties can be important, we refer to another study by Granovetter in which he surveyed a blue-collar worker in a suburb of Boston (Granovetter, 1974). Granovetter analyzed how people get their jobs and he found out that not just friends and family were the providers of information about a vacant job, but also acquaintance and loose connections. Even more, Granovetter concluded that the more a job is ranked superior the more information about this job opportunity is communicated via weak ties. This is the case because strong ties are often homogeneous ties (see homophily) while weak ties are connections to different people with different information, *Knowledge*, and *Resources*.

Dunbar's number. Robin Ian McDonald Dunbar leveraged the *Social Brain Hypothesis* (Dunbar, 1998), based on which the social group size of primates and humans is a result of the size of their brains. Dunbar analyzed the size of the neocortex (part of the brain that is responsible for higher functions, e.g. senses, spatial–temporal reasoning, and language) from different primates as well as their average group size and predicted a mean group size for humans (based on the size of our neocortex) of approximately 150. This number has since then been called *Dunbars's number* and was reconfirmed in empirical studies (Hill and Dunbar, 2003). Even some people have much more connections and others have less, on average this turned out to be a good estimation of social network sizes for humans.

Hierarchical organization of personal networks. In the previous paragraphs we talked about a) weak and strong ties and b) Dunbar's number. Consequently, the question arises how these 150 people are organized in different groups of tie strength. Zhou et al. (2005) answered this question with four hierarchical layers surrounding every human being (see Figure 2.2), with every group being roughly three times larger than the previous one. These groups are defined as follows:

1. *Support clique*. The strongest ties of a person including the core family

DRAFT

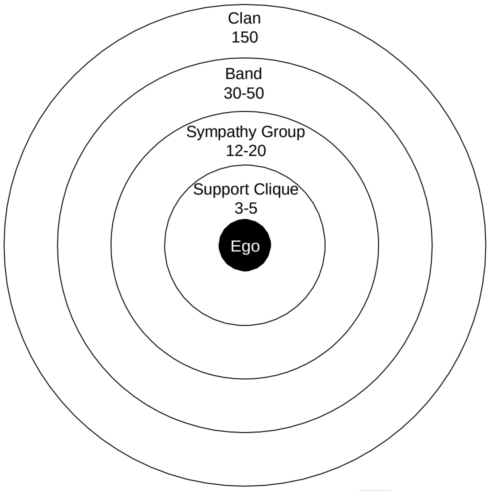


Figure 2.2: Hierarchical organization of nodes in a personal network (Zhou et al., 2005).

DRAFT

and the best friends is called support clique. In network studies these alters are often collected by asking the interviewee about the people with whom they "discuss an important personal matter". (Burt, 1984, p. 331). Size: 3–5 people

2. *Sympathy group*. The second layer consists of people that we have special relationships with based on regular interaction, e.g. most important co-workers, friends that we spend leisure time or share other common interests with. Size: 12–20 people.

3. *Band*. The third group are people that you interact from time to time but you don't feel emotionally very close. These are the people that you would also invite to your birthday party. Size: 30–50 people.

4. *Clan*. The largest layer includes the *active network* and includes "alters that ego feels they have a personal relationship with, and make a conscious effort to keep in contact with, or alters whom ego has contacted within the last 2 years" Roberts et al. (2009, p. 138) call this layer the *active network*. Size: 150 people.

Group Homophily. We now look at the group level of people interacting with each other (e.g. group of friends or co-workers). On the group level of social connections we repeat the theory of homophily. We talked about one person connecting to another person because of similarities. If we imagine that all human beings tend to connect with similar others, then this results in groups of homogeneous people:

> “Homophily in race and ethnicity creates the strongest divides in our personal environments, with age, religion, education, occupation, and gender following in roughly that order.” (McPherson et al., 2001, p. 415)

Six degrees of separation. The last two theories in this section describe the structures that are created by people on society level (e.g. a city, country, or the entire human race). In 1929, Karinthy (1929) wrote a non-scientific short story called “Chains” that became the first description of the global *connectedness* of all the people on planet earth. In Karinthys story one actor brings forward the argument that “Planet Earth has never bees as tiny as it is now” (reprinted in Newman et al., 2006, p. 21). He continues by enumerating chains of social connections between people that could connect any two people on earth with each other with five connections or less. Forty years later Travers and Milgram (1969) became famous with an experimental study on the same idea calling the phenomenon “Six Degrees of Separation”.

Small world. “It’s a small world” is an old saying. This refers to the fact described in the previous paragraph that everybody on earth is somehow connected to everyone else in short distances. At the same time, we also know that as a result of *homophily* and *transitivity* our personal networks are very densely knitted. When it comes to describing how people at large scale are connected with each other via interpersonal interactions, these two opposed characteristics are important (Hamill and Gilbert, 2009). So, on the one side we all know each other indirectly via a very few number of intermediates, but on the other side we just interact with 150 people on average and a lot of them are also connected with each other. How is it possible? The answer is *global connections*. Beside all our local transitive connections, every one of us has at least a small number of ties into different social worlds, e.g. to people in other countries, other cultures. And these *global connections*—often weak ties—tie mankind closely together. Watts and Strogatz (1998) developed an algorithm to create these small world networks artificially, but we will discuss issues of random and stylized networks later in this book. They also introduced the *clustering coefficient* that can be used to measure the amount of transitivity in a network. The clustering coefficient of a node is the percentage of possible links between its neighbors that are actually established. The clustering coefficient of a network is the average clustering coefficient of all nodes.

DRAFT

2.4 Analyzing social networks

Now that we know what drives the creation of links between *Agents* and which kind of network structures are formed by these dynamics, we will start to actually analyze networks. First, we visualize the connections of our network to get an impression of the underlying structure. Second, we are interested in identifying important actors in our networks. For accomplishing these points, use social networks created from the Tragedy of Julius Caesar. We discuss some details of creating these networks in this chapter and you can find the network matrices for all five acts in the appendix of this book. So, let's have a look at the Julius Caesar networks.

2.4.1 Visualizing networks

We spent a lot of time discussing different aspects of networks in the last couple of pages. Now it is time to actually see how networks look like when we visualize them. The matrix in Table 2.2 is easy to read but hard to interpret, e.g. it is not easy to identify groups of interacting people or the structure of this network. To overcome these drawbacks of the matrix representation, the sociogram representation is used. Sociogram pictures, which were developed in the 1930s by Jacob Levy Moreno (1953) are also called "stick-and-ball" diagrams. Looking at Figure 4.1 helps to explain why. Every *Agent* in our network is represented by a circle while every connection creates a line between two circles. The "x" in the matrix between the Soothsayer and Caesar is symbolized with a line between those two *Agents*.

This network visualization makes it easier to see the structure of the underlying network. We can see that the two tribunes Flavius and Murellus are interacting with citizens, that Caesar and the four senators Brutus, Cassius, Casca, and Antony form the core of this network, and that other individuals of the story are connected by a single line with this network core.

A network picture can help us to see an overview of the network structure, but it is just the first step of analysis. The decision of deciding which node of the network to put in a certain place in order to show a good network is, fortunately, not your decision. Special algorithms—so called layout algorithms (Torgerson, 1952; Fruchterman and Reingold, 1991)—are included in network analysis software to optimize the positioning of the nodes based on the network structure. We do not discuss further details of these algorithms in this book, but their purpose is simple. Connected nodes are positioned closely together, while unconnected nodes are kept apart. By doing so, the underlying structure

DRAFT

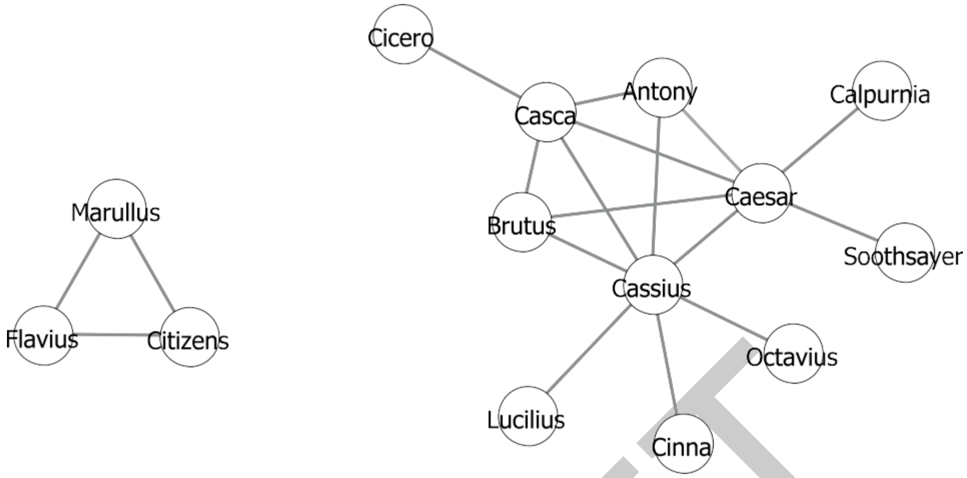


Figure 2.3: Network visualization of the first act of Julius Caesar

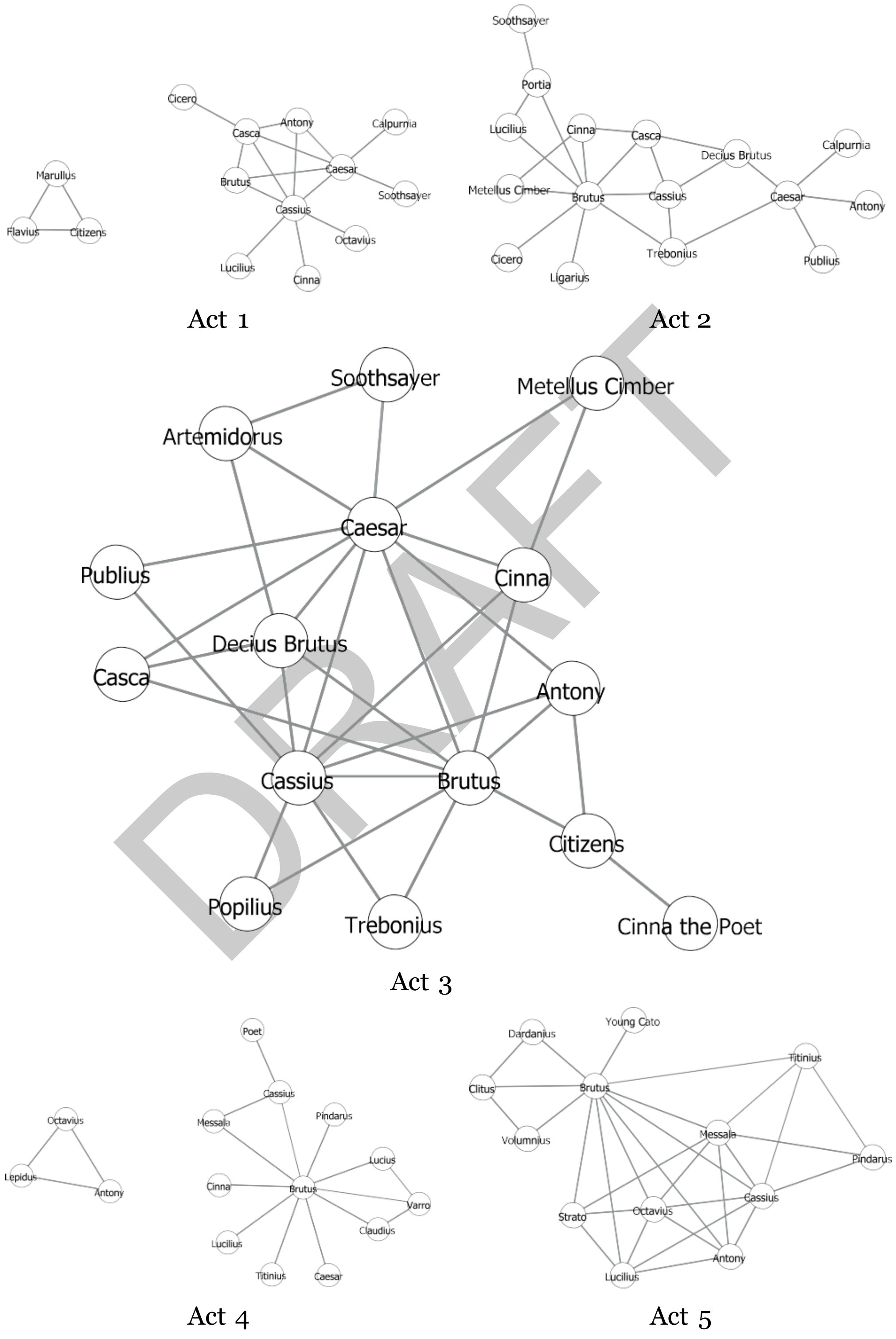
DRAFT

of the network can be revealed automatically.

2.4.2 Networks over time

One aspect of DNA is time. Time is normally included in networks by creating different networks for different time periods. In our Julius Caesar Network, every act of the play is represented by one network. The single scenes of an act are not identified separately, but they are aggregated to an act. Therefore, we have five different networks because the tragedy has five acts. In Figure 2.4 you can see visualizations of these five networks. The first picture is the same picture of Act 1 that we already know from Figure 4.1. The picture in Act 2 shows the increasing focus of the conspirators on Brutus while Caesar and Antony move to the periphery of the network. In Act 3 the many links between the *Agents* results in a densely connected network. The fourth picture is dominated by the divide of the main characters in two groups. The group with Brutus is on one side, and Antony is with the triumvirate on the other side. Brutus' group is larger in our network because Shakespeares play focuses on this character. The fifth picture shows us the clash of the two groups as well as the addition of some new *Agents*.

You can see that these five pictures are worth a thousand words. We see different parts of the story by looking at the visualizations of the network. Looking at the pictures is just the first step of the analysis. Of course, we can also do a lot more sophisticated analysis by calculating lots of different



Act 1

Act 2

Act 3

Act 4

Act 5

DRAFT

Figure 2.4: Networks of five acts of Julius Caesar

measures. In later chapters, you will even learn how to analyze not just one network but all networks together to statistically reveal change over time.

2.4.3 Identifying important agents

Once you have a network, the graph representation of a bunch of nodes, and the links between them, you can begin to identify characteristics of that network. One of the first things to consider is that some nodes stand out. Julius Caesar, for example, was connected to more people in ancient Rome than the Soothsayer. We know this as a result of knowing history, of reading the Shakespeare tragedy, or of reading the short introduction in chapter 1 of this book. We could also look at the graphical representation of the networks on the previous pages to figure out that Caesar must have been more important than other agents–at least unless he got killed. In case of larger networks, e.g. with hundreds or even hundreds of thousands of nodes, however, we had problems in identifying the important agents of the network by just looking at the data or at the visualization. Instead, we use network measures.

A measure is an algorithmic function that tells us something insightful about a network. In some ways, DNA is built upon the ability to apply measures to a complex network model and draw conclusions from those measures. There are a large number of measures that identify which things in a network are important or key. The set we are concerned with, at least initially, are those that measure the extent to which a node is of *central* importance. We are going to learn three measures and apply them to our Julius Caesar networks.

In 1979, the network researcher Linton C. Freeman created his conceptual clarifications of centrality in social networks (Freeman, 1979). Freeman identified "three distinct intuitive conceptions of centrality." In the following, we will introduce these concepts which are very important because the three measures based on these concepts are widely-used nowadays and also a lot of other measures are based on the fundamental ideas of these concepts. To illustrate these concepts we use a network which is similar to Freeman's network. In Figure 2.5 a star-like network is visualized. A single agent in the middle of the star, *Agent 1*, is connected to the other 5 agents while these agents are unconnected among them. Looking at this picture and trying to figure out why *Agent 1* is more important than the other agents in this network, leads to three different answers. These are the three concepts that Freeman was talking about:

1. *Agent 1* has more connections to other nodes than any other node in the network (degree)

DRAFT

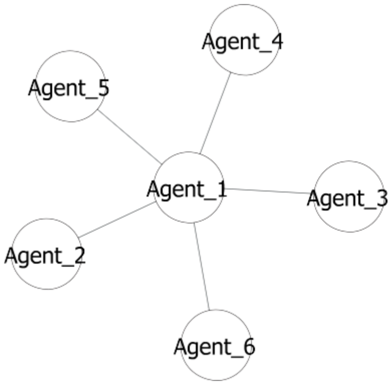


Figure 2.5: A simple network to illustrate different aspects of centrality

2. *Agent 1* has shorter distances to all other nodes than any other node in the network (closeness)

3. *Agent 1* is often in between–on paths connecting pairs of nodes (betweenness)

Degree centrality. The first concept is covered by *degree centrality*. *Degree Centrality* is a measure that tells the network analyst how many other entities are connected to the entities we care about. The assumption is that an agent who is connected to a lot of other agents, must be important. A high *Degree Centrality* is an indicator for an agent who is very active and therefore has a lot of connections. In our star network, agent 1 has a *Degree Centrality* of 5, while all other agents have a *Degree Centrality* of 1. In the model of our Julius Caesar Network, how many people is Julius Caesar, or any other agent, connected with? We run the measure *Degree Centrality* and discover who is the most connected entity, i.e. the most important *who* in this network. Will we be surprised?

Figure 2.6 shows the result for six selected agents of our network over time. It is obvious that Brutus is not very important at the beginning of the play but gets the agent with the most connection in the second part when he is selected to play a crucial role in the conspiracy. You can also see that some lines stop after three or four acts. Casca, who is the conspirator who stabs Caesar first from behind, is not mentioned again after the third act and Shakespeares text implies that he gets killed by the citizens after the assassination of Caesar. The opposite is true for Octavius who is just part of the network in the last two acts. The ghost of Caesar appearing to Brutus results in a *Degree Centrality* of 1.0 for Caesar after his death.

DRAFT

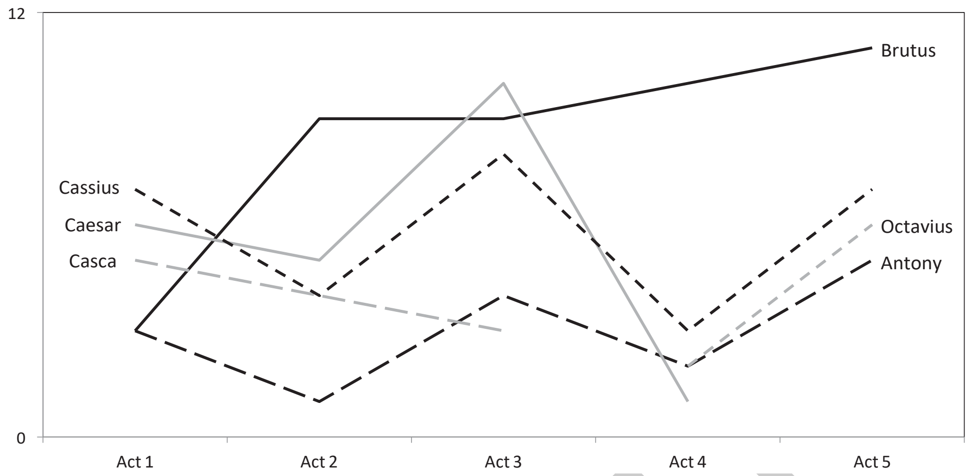


Figure 2.6: Degree centrality of six agents over the course of the tragedy

DRAFT

Closeness centrality. The second concept of centrality is *Closeness Centrality*. This is a measure representing the distance of an agent to all other agents in the network. To actually calculate *Closeness Centrality*, we first calculate the farness (Sabidussi, 1966). Referring again to Figure 2.5, we can see that *Agent 1* needs 1 step to all other 5 nodes resulting in a farness of 5. For *Agent 2* (as well as all other agents) the farness is 9 because of 1 step to agent 1 and 2 steps to the 4 other agents. Therefore, the more central an agent is, the lower is his distance to all other agents. And "distance" means "path distance" in the way it was introduced in the previous section. To calculate closeness centrality we use the inverse farness. Now, higher values point at the central agents of a network. Being close to all other nodes of the network is important because in that case communication paths are short and efficient and the access to different *Knowledge* and *Resources*of the agents is easier.

Betweenness centrality. *Betweenness* tells us which agents are important for the flow of communication. In terms that are more mathematical, *Betweenness* measures the number of times that connections must pass through a single individual in order to be connected. This measure indicates the extent that an individual is a broker of indirect connections between all others in network, similar to a gatekeeper for information flow in an organization. We can see that in the Julius Caesar Network such information would be highly valuable to Caesar. People that occur on many shortest paths between other people have higher *Betweenness* than those that do not. *Betweenness Centrality* is one of the key measures used by those interested in networks.

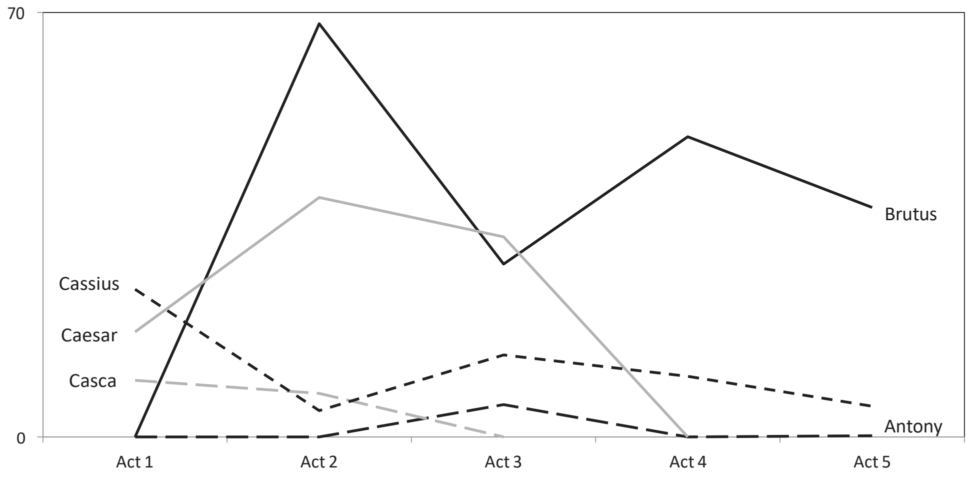


Figure 2.7: Change of betweenness centrality over time

DRAFT

In Figure 2.7 *Betweenness Centrality* is calculated for some agents. The most astonishing result is the big change of the score of Brutus over time. In act 1, Brutus is very unimportant for the information flow; he is just connected to agents that are also connected with each other and he lies on no single shortest path between two other nodes. This situation changes completely in act 2 when Brutus acts as an intermediate in many connections. When searching the shortest paths between all pairs of other agents, Brutus lies on these paths in almost 70 cases.

For all three measures which were introduced in the previous paragraphs you need to know one additional point. Look at the y-axes in Figure 2.6 and Figure 2.7. The values for *Degree Centrality* are in the range from 0 to 11 and even higher for *Betweenness Centrality*. These values are pretty much depending on the number of agents in the network. For example, in act 2 our network consists of 16 agents. If we want to know how often Brutus is on shortest paths between two other agents, we would look at 15 agents each of which connected to all other 14 agents resulting in 105 different pairs of agents. If we add just two nodes, this number goes up to 136. And the chance is quite high that Brutus, because of his central position in the network, would be on the shortest path of a large number of these pairs of agents. To prevent this artifact, we *scale* network measures with the maximal possible score. We do not discuss the scaling in detail at this point in the book . The interested reader can look in the glossary. For now, we state that the result of scaling centrality measures is that all scores are in the range of 0 and 1 for all networks.

2.5 Problem set

1. Think of your three best friends, and now of the three people you have been interacting with the most during the last 10 days. Do you see how simple definitions of your network can change your whole network data?
2. Why would you not add “Citizens” to your own network?
3. Describe *reciprocity* and *transitivity*. What is the main difference between these two concepts?
4. Why does *homophily* constraint the information that you get from your personal network?
5. Why is the world *small* after all regards to social distances?
6. What do the lines in the network visualizations represent?
7. Why does it make sense to analyze networks over time?
8. What is the highest possible degree one agent can have in a network with 50 agents?
9. What is meant by *sympathy group* and how big is your personal *sympathy group*?
10. Find one agent in act 3 with a betweenness centrality score of 0.
11. *Find five agents in act 3 with a betweenness centrality score of 0.
12. *Calculate the *farness* of the Soothsayer in act 2.
13. *Find a pretty network visualization online. Why do you like the picture? Is the network graph a good approach to visualize the underlying information? What is bad/confusing in your visualization?
14. *Cinna the Poet is getting killed by being confounded with Cinna the conspirator. Think of the centrality measures, what happens when you unify two actors to one node of your network?
15. *Explore the Julius Caesar networks from the book’s website. Create network visualizations that are similar to the figures in this chapter.

DRAFT

16. *Calculate different centrality measures and discuss the results.

17. **Read the centrality article by Freeman (1979). Freeman discusses his three measures exclusively for connected, undirected, and unweighted networks. Discuss implications for the measure calculation in the context of unconnected, directed, and weighted networks.

18. **Find out what about the average number of connections and the clustering coefficient of the Facebook friendship network. Imagine two random people A and B that are friends on Facebook. What is the expected number of shared friends of A and B?

19. **Gather the data for a connected and directed network with at least 15 nodes. You are now interested in who is connected to all other nodes on short paths. Which metrics X of the three Freeman metrics will you use? Why is X not trivial for your directed network? Based on the concepts of this metrics, create two new metrics X_{in} and X_{out} that are applicable for directed networks. Calculate the results of these metrics for your network and discuss the differences of the results.

20. **Perform a literature review to answer these questions: What are criteria for "good" network visualizations. Discuss the network pictures of this chapter with these criteria. Create a visualization of the third act that is *better* than our visualization. Why is it *better*?

DRAFT

DRAFT

Chapter 3

Meta-Networks

DRAFT

Based on the techniques Julius Caesar has learned in the previous chapter of this book, he can now complete a thorough network annual analysis of how all of his senators, generals and administrators *connect* to each other so he can figure out if there are any within his own imperial ranks that are in very central network positions and can become maybe a threat for himself—information that would be valuable to any dictator right? Julius Caesar is interested in a local rumored plot operating in the Roman imperial Senate. Caesar has identified several social networks containing a number of ”persons of interest” who communicate with each other regularly by letter, chatting at parades, conversing in whispers, meeting in secrecy. Based on carefully obtained surveillance Caesar constructs a network model of who is talking to whom and come up with an elaborate map detailing these relationships. He carefully analyzes this data and draws conclusions about how best to disrupt this network. Once again, it is the hope of the network analyst inside Caesar that the data obtained reveals vulnerabilities within the network structure.

To get better insights into his empire, Caesar conducts a survey (let us assume for now that all the important people in the empire participate in this survey and answer honestly). Caesar can soon discover from the data *who* is talking to *whom* inside his empire but he can't quite determine if by nature of these connections any of these pairs of communicators are a threat. After all, his model tells him very little about what they are talking about. His model tells him very little about what *Tasks* those that are talking share. His model tells him very little about what *Events* those who are talking share in common. Moreover, it would be nice if Caesar had a model of how the *Knowledge* of those who are assigned for specific *Tasks* actually fits the *Knowledge* needs to successfully accomplish these *Tasks*. Such a complex picture might indeed be what is needed to save Caesar from the Senatorial daggers.

In addition, Caesar is maybe interested in which armories are accessible by which generals within his empire. He studies in close detail an inventory of the all the armories as well as the servers they are networked to and how they communicate with each other. He wants to take this information and draw certain conclusions about the way the network is structured so he can draw conclusions about how to make his military stronger. Will his current network model of how the armories are connected help him get the job done?

In all these examples we can get network data about people and other entities are connected with each other, be they workers conspiring senators in the Roman imperial senate, administrators and his trusted generals in the upper echelons of his government or how his armories are all interconnected. Clearly the complex information described above would prove extremely useful in providing an in-depth analysis of the dynamics relevant to Caesar's dictatorship. *Meta-networks* are the models to describe this complexity in a network model.

The set of networks for two or more of these entity classes is called the meta-network. The entire field of Dynamic Network Analysis (DNA) is based on the concept of the meta-network. Therefore, we need to ask what is a meta-network, and how does it relate to our Julius Caesar dataset? In this chapter, you will learn about the different node classes beyond *Agents* and how they can be used and combined to create various networks. We will use these new node classes to collect even more data from the Tragedy of Julius Caesar. And you will learn about different concepts and measures which make use of all these networks.

DRAFT

To begin with, what is an *entity*? In the previous chapter we defined network entities as human beings. In this chapter we are going to extend this perspective. An entity is essentially the building block of all networks. In general, an entity is a node in a network. It is what we are networking. It is what we are looking at and it can literally be anything that you can possibly think of in terms of what you can possibly imagine. You name it–it is an entity! We can't build a network without entities. In DNA, an entity is often best described as a *who*, *what*, *where*, *when*, *how* or *why*. These descriptions are the hallmark of any good news story and are convenient ways to describe any complex system or story. Networks tell stories and vice versa. We can almost extrapolate any network from a story and a network model may indeed tell a story, but we are getting ahead of ourselves.

If you take a few moments to ponder this, most anything you can think of can fit into one of these categories. So, think of entities in terms of *who*, *what*, *where*, *when*, and *why*:

- A *who* such as Julius Caesar, Cassius, Brutus, CEOs, famous historical people, imaginary people, myths, your friends, family members, terrorists, people that owe you money, scientists, celebrities, athletes, religious figures, etc.
- A *where* such as The Roman Senate, The streets of Rome, Brutus' House, cities, stores, swimming pools, lakes, rivers, oceans, countries, roads, Planets, galaxies, etc.
- A *what* such as a dagger, computers, satellites, cars, cell phones, food, money, molecules, etc.
- A *why* Julius Caesar is becoming too powerful and so should be killed, other beliefs and attitudes.

So now that we met the *whos* in Julius Caesar, remember that an entity can be literally anything. But for a moment, think of all the connections that would compose a network of a company or a country. You couldn't describe these networks with one type of entity called "nodes". If indeed it were possible at all, it is plainly obvious that we would need to interconnect all sorts of different types of entities. We would need a model beyond a mere network of same type entities. We would need a better model, one that would incorporate different entity classes and allow us to perform an analysis on the model that way. Such a model is called a meta-network.

Before we talk about meta-networks, we to need get a firm understanding of what an entity is and how entities are the building blocks of networks. In this chapter, we want to create a meta-network of Julius Caesar, which will attempt to capture and present to us a model of all the entities that make up the plot of Shakespeare's classic play. Even though it may seem small, with only 36 *whos*, just watch and see what happens when we start factoring in *Locations*, *Knowledge* bases, *Events*–every entity-types that can meaningfully describe a *who*, *what*, *when*, *where* and *why* of the Julius Caesar Network model. Indeed, things can get complex fast as you will also come to see. And of course, real world meta-networks can have tens of thousands of nodes.

3.1 More than *who*: Additional entity classes

It is useful to classify certain type of *entities* into categories or classes. A group of entities of the same type is referred to as an *entity class*. The relation of

DRAFT

entity class to entities is sort of like genus to species in the Linnaeus system of classification. Toward those ends, there are ten entity classes that social scientist have determined to be of the most value in network analysis: *Agents*, *Resources*, *Knowledge*, and *Tasks* are the main node classes for which most of the meta-network measures have been developed (see in the next sections of this chapter). The other node classes are *Organizations*, *Locations*, *Events*, *Actions*, *Beliefs*, and *Roles*. These are the genus categories of which we can fit most species–entities. Are they beginning to sound familiar now? Can you see the parallel between these types and the *who*, *what*, *when*, *where*, *why* and *how* model?

These classifications are what drive many of the advanced mathematical algorithms that can make a highly complex meta-network comprehensible to an analyst. The good news is that nearly anything that can be an entity can neatly fall into one of these entity classes and you will see that as we build the Julius Caesar model, we will place our entities into such containers as the entity class types described above. In detail, the ten entity classes can be defined as follows:

1. *Agents* are individual decision makers. The most common type of decision makers are people. However, this category could also be used for other types of actors such as robots or monkeys. A single *Agent* represents any person: a family member, a Roman soldier, Soothsayer, Calpurnia, Cicero, any historical figure, a terrorist, or a teacher.

2. *Resources* are products, materials, or goods that are necessary to perform *Tasks*, *Events*, and *Actions*. A *Resource* could literally be a dagger, a cloak, a crown, a short sword, a computer, money, bombs, tools, or books.

3. *Knowledge* describes cognitive capabilities and skills. *Knowledge* could be Trigonometry, History, English, Economics, the science of DNA, or the *Knowledge* about how to perform a surgery or to build bombs.

4. *Tasks* are well defined procedures or goals. A *Task* could be any process in a company, e.g. product development or administration, but also the plan to kill Caesar.

5. *Organizations* are collectives of people that try to reach a common goal. An *Organization* could be a specific company, the United Nations, or the government of a country.

DRAFT

6. *Locations* are geographical positions at any aggregation level that describe places or areas. A *Location* could be 1600 Pennsylvania Avenue in Washington, London, Florida, The Middle East, Earth, or Mars.

7. *Events* are occurrences or phenomena that happen. An *Event* could be 9-11, the JFK Assignation, the Super Bowl, a wedding, a funeral, or an inauguration. Specific *Events* are one time occurrences with a specific date.

8. *Actions* are specific activities done by *Agents*. An *Action* could be buying a car, writing a letter of recommendation, or flying to Africa.

9. *Beliefs* are any form of religion or other persuasion. A *Belief* could be to believe that there is a god, or that there are many gods, or that that Earth is flat. Some *Beliefs* are signaled by sentiment such as ”war is bad.”

10. *Roles* describe functions of individual decision makers abstracted from specific *Agents*. A *Role* could be leader of a group, driver of a car, or mother of an *Agent*.

In addition to these 10 node classes we use an 11*th* node class that is different. *Groups* also referred to as meta-nodes, are any categorization of nodes into a cluster. *Groups* are defined by one or more of the grouping algorithms (see next chapter).

To repeat the function of these node classes and to show their differences as well as give a first impression about how these node classes are connected with each other, we summarize the Julius Caesar plot very briefly. Brutus (*Agent*) and other senators (*Roles*) agree that Rome (*Location*) would be a better place without Caesar (*Belief*). To kill Caesar (*Task*) they form a group of assassins (*Organization*). To accomplish their *Task* they need to know about Caesar's daily routine (*Knowledge*) and how to get their knives (*Resources*) into the senate. Finally, the assassination (*Event*) takes place because somebody actually stabs Caesar (Action).

So when building a network the DNA scientist needs to be aware of how to categorize the entities he or she wishes to study, that is a decision has to be made as to what entity classes are needed and where it makes sense to put the entity in. To a certain extent one can make up their own entity classes sole and separate from the ones aforementioned. However, it probably is not wise to do so; by fitting your entities into a traditional, if not obvious, entity classes, we can then run powerful measures on them. What is a measure? We will get to that shortly.

DRAFT

The networks in chapter 2 were all just constructed by using nodes from one single node class—*Agents*. You have just learned that there are ten different node classes. When combining them with each other, we can create a lot of networks. Most of these *new* network data are different from the *Agent* x *Agent* networks of chapter 2. For instance, a matrix describing the connections between *Agents* and *Knowledge*, i.e. a skills network, is not squared but rectangular as the x-axis (*Agents*) and the y-axis (*Knowledge*) have different entities. If you are having a hard time to imagining these rectangular matrices, don't worry, you will see a lot of them at the following pages.

But before we can come to the point of coding the meta-network of the Tragedy of Julius Caesar, we first have to make the decision which entity class we will need. Naturally, we need to think up of a least two different networks. But, to really hammer the point home, let us come up with three, maybe five different networks relating to Julius Caesar and add them together to see what we get. Again, before we truly get into DNA, our goal is to create multiple networks to create our multi-modal "meta-network." So, let's now create another network by repeating the process for building our *who* by *who* network and then another until we are satisfied that we have captured enough pertinent and interesting data that could be of use to us if presented as a meta-network. For purposes of our Julius Caesar model, we are going to capture about as many useful networks as possible, graphing the relationships of *who*, *what*, *where*, *how*. The *when* is covered by the temporal aspect of the five acts which have been introduced in the previous chapter. To cover the other aspects, we will create the node classes *Knowledge*, *Tasks*, *Locations*, and *Events*.

DRAFT

3.1.1 Skills in the roman empire

To build our first network with nodes other than agents, we are going to graph *who* x *what*. The *what* in this case represents *Knowledge* (K). So, our *who* by *what* network constitutes an *Agent* (A) by *Knowledge* (K) network. Let us begin by determining a list of what *Knowledge* bases would seem applicable to our network. We defined Knowledge as skills. Based on our understanding of the play, we have identified the following *Knowledge* sets to be of relevance to the network analysis. Table 3.1 enumerates the six *Knowledge* nodes that we have identified in the Tragedy.

We just selected the—from our perspective—most important skills. Of course, you could come up with a totally different list. In networks that are derived from real world *Organizations*, the list of *Knowledge* entities can be

Administration	Persuasion
Citizenry	Politics
Military	Prediction

Table 3.1: Knowledge list in Julius Caesar

surveyed from the members of the *Organization*. In companies with *Knowledge Management* lists about *who* is connected to which *Knowledge* is often pre-collected and can be directly transferred to a network matrix.

For now we focus on the selected skills of the Julius Caesar network. The next step, after identifying the list of *Knowledge*, is to connect the *Agents* of our meta-network with these *Knowledge* entities. Figure D.1 in the appendix shows the network matrix of these connections. An "X" in this matrix, e.g. between Anthony and Persuasion, indicates that we think that the *Agent* Antony has the *Knowledge* Persuasion. In case our network matrix would be more elaborated, we could code weights into the matrix. What does this mean? Instead of "X" we would use numbers, for instance, from 1 (little skills) to 5 (very high skills). Why would this be a useful extension to our network matrix? The answer lies in the story. When you remember the plot of the Tragedy of Julius Caesar, Brutus has the *Knowledge* of persuasion, which is important to manipulate the Citizens after the assassination of Caesar. But Antony, who speaks after Brutus to the Citizens has higher skill in Persuasion. Finally, Antony is able to convince the Citizens to support his cause. Therefore, we could code Brutus with "3" and Antony with "5" to express this difference. But for the purpose discussing the node classes, we omit this information to keep the networks simple. For your own data collection process it is probably worth spending some time with considering whether the additional work of coding and handling weighted *Knowledge* (and also other) networks pays off or not.

3.1.2 Tasks that drive the tragedy

Now let us create *who* x *how*, or *Agent* (A) by *Task* (T) network. Just like in the previous examples, here is what we have determined to be the *Tasks* as we took them to be by studying the plot of Julius Caesar in detail. Table 3.2 enumerates all *Tasks* of the tragedy. Once again, we have created a list, much like our previous *Knowledge* sets, based on our personal insights into the Julius Caesar Network. In the appendix you can find the *Agent* x *Knowledge* network matrix.

But now, read the list of *Tasks* in Table 3.2. Do you agree with this list

DRAFT

Achieve Victory	Kill Caesar
Attend Senate	Lure Caesar
Avenge Caesar	Persuade Brutus
Celebrate Victory	Persuade Caesar
Deceive Brutus	Persuade Citizens
Defeat Antony/Octavius	Read Will
Defeat Brutus/Cassius	Refute Brutus
Expunge Conspirators	Support Antony/Octavius
Form Coallition	Support Brutus/Cassius
Haunt Brutus	Warn Brutus
Justify Murder	Warn Caesar

Table 3.2: Task list in Julius Caesar

Battle Tents	Parade to Senate
Battlefields	Pompey Parade
Brutus' House	Senate
Funeral site	Streets of Rome

Table 3.3: Locations where the tragedy takes place

or do you have problems in accepting that some of these *Tasks* are really a *Task*? If you have doubts about the list, you are right. Some of these *Tasks* could probably also be part of different node classes. For instance, *Read Will* is maybe rather an *Action* than a *Task*. Our decision to code all these entities as *Tasks* is more than anything else based on the fact that we did not want to add another node class to the meta-network. There are, of course, better reasons for coding an entity as *Task* or as *Action*. *Tasks* are—and we will discuss this more in details later in this chapter—things that need to be done. You can see a *Task* as a goal. So, reading a will is a *Task* in case it is a goal that needs to be achieved, e.g. because it is a sub-goal of a bigger goal like *Gain Power in Rome*. Reading a will is an *Action* if it just happens without identifying this as an important goal. After these considerations, what do you think is *Read Will* for the Tragedy of Julius Caesar, *Task* or *Action*?

3.1.3 Where everything takes place

In our additional node class, let us decide to collect the places where the characters in The Tragedy of Julius Caesar have been seen. For a written document like a play it is quite easy to determine the list of *Locations* since every scene starts with the information where this scene takes place. Nevertheless, the decision remains whether a location is of relevance or not. We

DRAFT

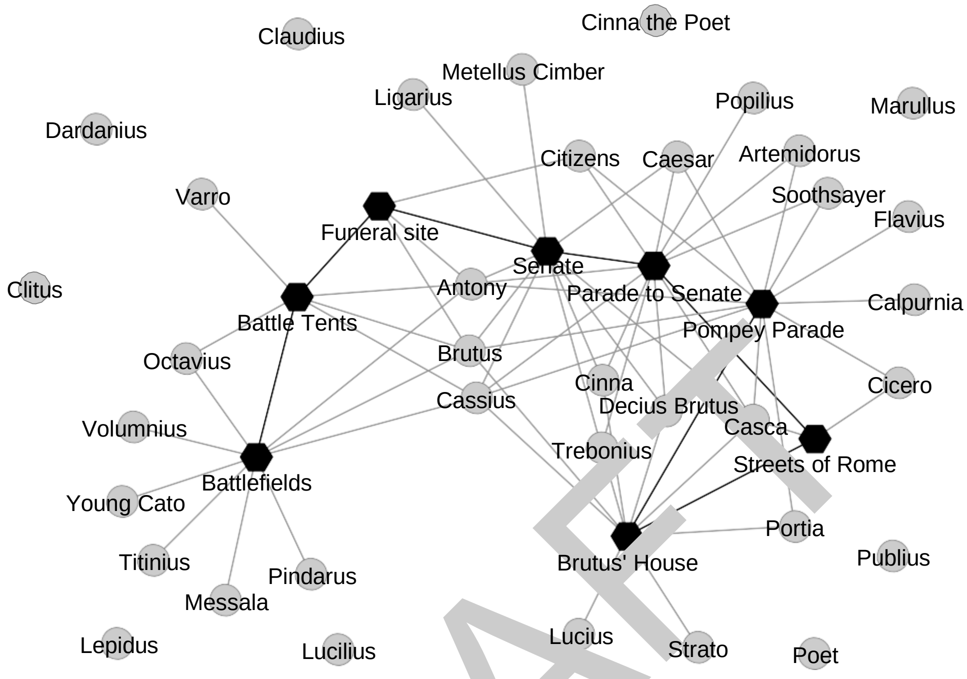


Figure 3.1: Agent and Location network

decided to code 8 locations (see Figure 3.3).

The network matrix that is created by connecting this node class with our *Agents* is the *where* x *who* network or simply put: who has been where. The matrix can be found in the appendix. What also can be found there is a *Location* x *Location* matrix. Since the *Locations* are ordered in a sequence based on the narrative of the story, we connected the *Locations* in the same order. In Figure 3.1 we visualized both networks in one picture. The black nodes (hexagons) are the *Location* entities. They are connected with black lines in the sequence of the story line, starting from the *Pompey Parade* and ending at the *Battlefields*. The gray circles are the *Agents* that are connected to these *Locations*. We can see in the visualization that the prior and the later *Locations* form two more cohesive groups in the right and left side of the visualization. From the *Agents* just Antony, Cassius, and Brutus connect both groups of *Locations* with each other. Therefore, the network layout algorithm put these *Agents* between the groups of nodes. Some minor *Agents* of the story are not connected to any *Location*. Their position in the visualization

DRAFT

Antony speaks to citizens
Antony/Octavious honor Brutus
Antony/Octavious victorious
Artemidorus attempts to warn Caesar
Battle, Pindarus fooled
Brutus kills himself
Brutus speaks to citizens
Brutus' defeated
Brutus/Cassius argue
Brutus/Cassius prepare for battle
Caesar warned by soothsayer
Calpurnia warns Caesar
Cassius orders Pindarus to kill him
Cassius/Brutus driving into exile
Cassius/Brutus raise armies
D. Brutus coax Caesar to attend Senate
Funeral of Julius Caesar
Generals meet on battlefield
Ghost of Caesar haunts Brutus
Meeting at Brutus' house
Octavious arrives in Rome
Octavious, Antony, Lepidus ally
Octavius and Antony march army
Offering of crown to Caesar, he rejects
Parade, celebrating defeat of Pompey
Parade, to Senate
Portia kills herself
Reading of Caesar's will
Rome plagued
Senate meeting, Caesar killed
Tintinius kills himself

Table 3.4: Events of the tragedy

is randomly selected in the periphery.

In the context of *Locations* you probably want to discuss issues of *Granularity*—the level of aggregation. For instance, what if we use a node for every battlefield instead of one aggregating all of them? Or where actually is *Streets of Rome*? Shouldn't we enumerate the different streets where different incidents happened? And if so, how different would the networks be? We do not answer these questions at this point but we will discuss this aspect of *Locations* in chapter 5. For now, be aware of the fact that different levels of aggregation result in different networks.

3.1.4 Events as corner posts of the story

The last node class that we introduce for the Julius Caesar meta-network is *Events*. We defined earlier in this chapter that *Events* are "are occurrences or phenomena that happen". And there are a lot of things that happen in a tragedy by William Shakespeare. From the perspective of the play, *Events* form the storyline. Table 3.4 enumerate all the *Events* that we have identified in the text. Once again, we can have the discussion about whether a single entity of this list is really an *Event* or rather an *Action*. We discuss this question more generally some pages later.

Moving on and creating networks, we collected data for three different networks including *Events*. For all of which you can find the network matrix

DRAFT

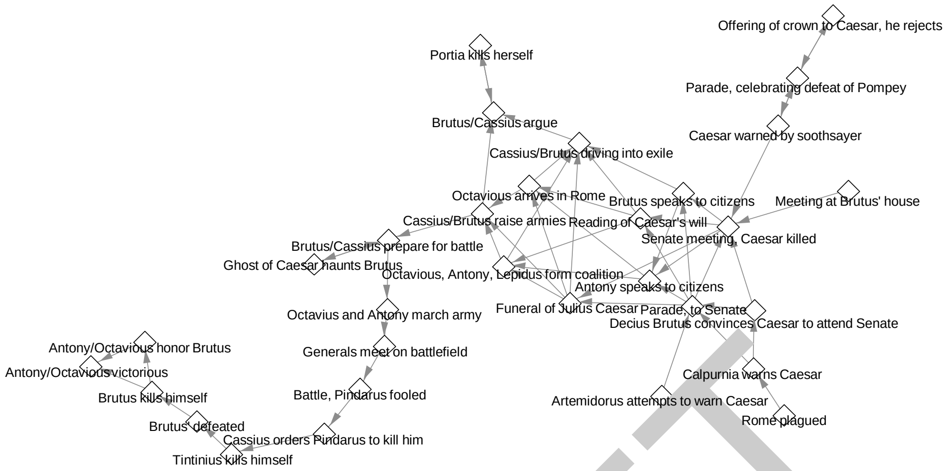

Offering of crown to Caesar, he rejects
Portia kills herself
Parade, celebrating defeat of Pompey
Brutus/Cassius argue
Caesar warned by soothsayer
Cassius/Brutus driving into exile
Octavious arrives in Rome
Brutus speaks to citizens
Meeting at Brutus' house
Cassius/Brutus raise armies
Reading of Caesar's will
Senate meeting, Caesar killed
Brutus/Cassius prepare for battle
Ghost of Caesar haunts Brutus
Octavious, Antony, Lepidus form coalition
Antony speaks to citizens
Funeral of Julius Caesar
Parade, to Senate
Octavius and Antony march army
Decius Brutus convinces Caesar to attend Senate
Antony/Octavious honor Brutus
Generals meet on battlefield
Antony/Octavious victorious
Calpurnia warns Caesar
Battle, Pindarus fooled
Brutus kills himself
Artemidorus attempts to warn Caesar
Rome plagued
Brutus defeated
Cassius orders Pindarus to kill him
Tintinius kills himself

Figure 3.2: Event x Event network of Events following each other

in the appendix. The first network is obvious, the *Agent* x *Event* network showing the participation of *Agents* at certain *Events*. The second *Event* related network connects the *Events* with the *Locations* where the *Events* took place. The third is another network that put entities in a temporal order; the *Event* x *Event* network connects those *Events* that follow each other in a temporal order. Figure 3.2 shows a visualization of this network and gives a short overview of the tragedy.

DRAFT

3.1.5 Some comments on node classes

Specific vs. generic nodes. You have learned a lot in this chapter about entities and how to organize them in different node classes. At the beginning of this chapter we introduced ten node classes. Four of which—*Agent*, *Organization*, *Event*, and *Location*—are special in a sense that they can be composed of two different types of nodes. Recall the discussion about what is an *Agent* and what is not an *Agent* in chapter 2. Do you remember our concerns about the node *Citizens*? What is the issue with this node? It is not *specific* but *generic*. Here is the difference between these two types of entities:

- *Specific entity.* A particular entity representing a uniquely identifiable *Agent*, *Organization*, *Event*, or *Location*, e.g. Barack Obama, IBM, Christmas 2012, Rome (Italy).
- *Generic entity.* A generalized entity representing an *Agent*, *Organiza-*

tion, *Event*, or *Location*, e.g. citizen, committee, Christmas, house.

It is importance to differentiate between generic and specific entities. This is particularly important when it comes to measure calculation with network data. We talk more about measures later in this chapter, but we can imagine a simple example: Let us assume that some police officers in Rome wanted to analyze the background stories behind the assassination of Julius Caesar some days after his death. For this purpose, police officers fan out to different *Locations* and try to figure out who participated in different *Events* that took place in these *Locations*. The result of this data gathering process could be similar to our *Agent* x *Location* network in Figure **??**. After analyzing this network, guess what, beside *Brutus* and *Antony*, the *Agent* “Citizens” is very important in this network. So, while the fact that citizens are connected to some incidents or to some *Agents* was an interesting information for visualizing the data and understanding the story, this becomes problematic in the context of measures because one and the same generic node can represent different specific entities. Therefore, a generic node can—dependent from the measure—create artifacts in the results.

Knowledge vs. Resources. Note that there is one entity set that we are not employing in our Julius Caesar dataset even this node class sounds quite important based on the definition at the beginning of this chapter, and that is *Resources*. Why have we opted not to use *Resources*? Primarily because the main characters in the tragedy did not use important *Resources* during the course of the play. In addition, it seemed that *Knowledge* was every bit as a *Resource* and they were in a sense interchangeable in our network. It is not always easy to decide whether to put an entity into a *Knowledge* node class or a *Resource* node class. For instance, having a Bachelor’s degree in computer science is connected to a lot of *Knowledge*. That Bachelor’s degree can surely be also a *Resource* for an IT department of a company.

The good news for you: Later in this chapter you will learn about different measures for meta-networks. When you read about these measures as well as their underlying concepts, you will figure out that *Knowledge* and *Resources* are most of the time treaded interchangeable by the network measures. Therefore, it is actually not very important whether you decide for *Knowledge* or *Resources*, as long as you stay consistent within your project.

Task vs. Action vs. Event. We also had some difficulties in the previous paragraphs to decide whether an entity is a *Task*, and *Event*, or an *Action*. For clarification, *Tasks* are scheduled or planned activities, for instance, from the perspective of the conspirators in William Shakespeare’s tragedy, killing Caesar is a *Task*. The assassination of Caesar is an *Event*—even more a

DRAFT

specific *Event*. Focusing more closely to the *Event*, we could have the need to describe a bit more in detail what is going on. For this reason we also have the node class *Action*. So, the actual killing—the stabbing of Caesar—can be coded as *Action*.

In general, you are very much interesting in *Tasks* as *Tasks* describe the regular patters of activity in a meta-network. In addition, you will see that *Tasks* are heavily involved in multi-mode measures which is also an argument for coding entities that are hard to decide rather as *Tasks* than as *Event* or *Action*.

3.2 Adding it all together – the meta-network

As you should be seeing by now, real networks are not one dimensional. People have Knowledge, they have access to Resources, Events happen and change networks and certain nodal points will contain varying attributes distinguishing them from the others. In fact, real networks for that matter are rarely one dimensional except, perhaps, on the most abstract levels. For the Julius Caesar network we have coded the following ten networks:

1. *Agent* x *Agent* (who knows who)
2. *Agent* x *Event* (who goes to what)
3. *Agent* x *Knowledge* (who knows what)
4. *Agent* x *Location* (who is where)
5. *Agent* x *Task* (who does what)
6. *Event* x *Event* (when by when)
7. *Knowledge* x *Task* (what is needed to do what)
8. *Location* x *Location* (where by where)
9. *Task* x *Event* (what by when)
10. *Task* x *Task* (what by what)

When looking at this list and recalling the ten node classes that we have introduced in the beginning of this chapter, it is easy to imagine that there are many more possible networks that we can create with these 10 node classes. Table 3.5 shows all networks based on all possible combinations of node classes. As you can see, there are 55 different networks for one single meta-network. Of course, in most of your network projects you will not use them all at the same time, but be aware of them because every additional network can give you deeper insights into your real-world system that you are going to describe.

DRAFT

Now you may be saying to yourself–just what kind of model are we now left with? This complex meta-network could be said to resemble a ball of yarn. What are we to make of it? Well, your reaction is to be expected because meta-network analysis is beyond first intuitions and the resultant meta-network can be hard for an analyst to comprehend. However, this is where powerful computational methods come into play in the form of measures and we can begin to break down the meta-network and glean some powerful insight into the network's architecture.

3.3 Concepts for two entity classes

DRAFT

It is time to revisit measures once again. We learned in previous chapters that a measure tells us something unique about a network using computational methods. More specifically, a measure is an algorithm specially formulated to tell us meaningful information about network data that we apply our algorithm to. In chapter 2, you have learned about centrality measures that can identify important nodes in networks.

So far as the dynamic network analyst is concerned, measures can span two or even more entity classes. Such measures are carefully constructed and arrived at by research and scientific method.

Measures and algorithm are created to answer questions by analyzing network data. These questions can be grouped based on the underlying ideas. We call these groups of measures and questions *concepts*. In the following we describe these concepts and introduce some measures that can be derived from the concepts. In this chapter we focus on discussing the measures. We will also use the measures to analyze the Julius Caesar network. We will not discuss the algorithmic details of the measures. This will be accomplished in the appendix. The interested reader can explore all the mathematical details of the measures that we are using at the end of this book.

3.3.1 Quantity

Let's start with the easiest and most intuitive concept of two-mode measures—quantity. Quantity measures simply count or summarize information. Remember *Degree Centrality* (Freeman, 1979), a measures that we used in chapter 2 to identify the most active *Agents* in our network. This centrality was calculated in the one-mode network describing the interactions between *Agents*. We use the same idea to describe activity in two-mode networks. We constructed the *Agent* x *Task* network earlier in this chapter. We can now use

DRAFT

	Agents	Knowledge	Resources	Tasks	Organization	Event	Actions	Locations	Beliefs	Roles
Agent	Interaction Network	Knowledge Network	AR Network	Assignment Network	Employment Network	AE Network	AA Network	AL Network	AB Network	AX Network
Knowledge		Information Network	KR Network	Requirement Network	Compe-tency Network	KE Network	KA Network	KL Network	KB Network	KX Network
Resources			RR Network	Precedence Network	Industrial Network	RE Network	RA Network	RL Network	RB Network	RX Network
Tasks					Interorganizational Network	TE Network	TA Network	TL Network	TB Network	TX Network
Organization						OE Network	OA Network	OL Network	OB Network	OX Network
Event						EE Network	EA Network	EL Network	EB Network	EX Network
Actions							AA Network	AL Network	AB Network	AX Network
Locations								LL Network	LB Network	LX Network
Beliefs									BB Network	BX Network
Roles										RX Network

Table 3.5: Meta-network matrix with all possible 55 networks based on 10 node classes

Group	Concept	Description
Quantity	Degree	Count row or column entries
	Load	Average link values/counts of network
Variance	Centralization	Distribution of node level scores
	Diversity	Concentration of node level scores
Correlation	Similarity	Degree of similarity between two *Agents*
	Distinctiveness	Complementary of two *Agents*
	Resemblance	Exact same connections
	Expertise	Degree of dissimilarity between *Agents*
Specialization	Exclusivity	*Agents* with exclusive connections
	Redundancy	Different *Agents* sharing connections
	Access	Identify critical connections

Table 3.6: Classification of measure concepts for two entity classes

this network to calculate the activity of *Agents* based on their assignment to *Tasks*. Counting the number of *Tasks* for ever *Agent* results in this list of *Agents* with connections to more than five *Tasks*:

1. Antony (15)
2. Cassius (13)
3. Brutus (11)
4. Decius Brutus (11)

Degree. The numbers in brackets behind the *Agents* represent the number of *Tasks* a single *Agent* is connected to. This number is hard to interpret; is 15 or 11 high or low? For instance, if the overall number of *Tasks* in our empire were 15, then Antony would be connected to every single *Task*. In opposite, if this number were 150, then Antony would be connected to just one-tenth of all *Tasks*. To make it easier to interpret *Degree Centrality* over different networks, network analysts scale the measure so that the result is in the range of 0 to 1. This can be done by dividing the degree (e.g. 15 for Antony) with the maximum possible degree which is the size of the *other* node class (Borgatti and Everett, 1997). As we have 22 *Tasks*, this is the number to scale our results. Therefore the scaled *Degree Centrality* of Antony is 0.68. In other words, Antony is connected to 68 % of all *Tasks*. This idea, of scaling by dividing with the maximum possible value to get a result between 0 and 1, is used for almost all network measures.

Load. Computing Degree Centrality results in a single value for every actor — or for every *Task* in case we count the number of *Agents* that are connected to the *Tasks*. Consequently, this measure is a *node level* measure. In opposite,

DRAFT

Network	N.1	N.2	Edges	Load	Density
Agent x Event	36	31	100	2.78	0.09
Agent x Knowledge	36	6	54	1.50	0.25
Agent x Location	36	8	60	1.67	0.21
Agent x Task	36	22	74	2.06	0.09

Table 3.7: Load and Density of different networks

a *network level* measure creates a single value for an entire network. An easy way to create a network level measure with quantitative information is to calculate the average number of *Tasks* (or *Knowledge*) per *Agent* or we could calculate the *Density* of the network matrix by looking for the proportion of connections compared to all possible connections between all *Agents* and *Tasks*. We call the concept of these quantitative network level measures *Load* since they measure how loaded a network matrix is with the edges. As for network level measures every network can be represented with a single value, this approach can be used to contrast different networks with each other.

Table 3.7 illustrates such calculations for four networks that include *Agents*. The second and third columns of the table show the number of nodes in the first and in the second mode of each network. The third column holds the number of edges in the networks. With these numbers, the network level measures Load and Density are calculated. You can see that on average every *Agent* of the play is connected to 2.78 *Events* and 2.06 *Tasks*, while these numbers are smaller for *Knowledge* and *Locations*. On the other hand, for the *Agent* x *Knowledge* and the *Agent* x *Location* networks the density is more than twofold than for the other two networks.

Beside the results regarding the content of the network, we can see a very important artifact when calculating Density in networks: The result value has an inverse dependency on the network size—the large the network, the lower the Density. This is true for all networks. Why is this the case? The answer can be found in the limitation of human connections. Imagine the number of people in your life, all of which you have personal interactions from time to time. Scientist figured out that this number is 150 on average (Hill and Dunbar, 2003). Let us assume that all these people live in the same city than you do, that this city has 150,000 inhabitants, and all the other people also have 150 connections within this city. If we now construct a network matrix for the *Agent* x *Agent* network of this city, this matrix has 22.5 billion cells — wow! In contrast, as every *Agent* has 150 connections, the number of ”X” in this matrix is ”just” 22.5 million resulting in a network density of 0.1%. If you now move under the same assumptions with your whole network to a small

DRAFT

town with 5,000 people, the density is suddenly 3.0%. So, density actually tells us something about the network, but you have to be careful with the interpretation when comparing networks with different size.

3.3.2 Variance

Centralization. A different perspective of analyzing two-mode networks on network level is to look at the distributions of the values on node level. For instance, some pages ago we've calculated Degree Centrality for the *Agent* x *Task* network and gave you a list with the top 4 scoring *Agents*. But what is the result of the other 34 actors? Is the load of work evenly distributed or are there some *Agents* that are involved in many *Tasks* while a lot of other *Agents* are way less active? Variance measure can help us to answer this question. Freeman (1979, p. 227) stated that "the centrality of an entire network should index the tendency of a single point to be more central than all other points in the network." Therefore, if a single node — or some nodes — has a much higher centrality score than all other nodes, then the *network centralization* is high. And with "high" we mean close to 1.0 which is — due to scaling — the highest possible centralization value. In case the centrality scores are almost evenly distributed, the centralization value tends towards 0.

Highly centralized networks are often not very robust since single nodes have very much power, influence, or control over the network and removing these nodes can endanger the whole network. So, if Caesar would calculate different centrality measures to identify the key players in his empire, he could also calculate centralization measures to get a better understanding of how dependent the network is on these central *Agents*. looking at Table 3.8 we can see that, even though the networks are very different in size and maximum degree, the centralization is very similar. The higher centralization of the *Agent x Location* network is a result of the fact that a small number of *Agents* showing up in almost every relevant *Location* (e.g. Brutus, Antony) while a lot of other *Agents* just occur in one or two *Locations*.

Diversity. Another measure of *Variance* is *Diversity*. Many diversity measures were developed in Ecology (Magurran, 2003) where scientist are interested in measures that tell them whether an ecosystem is dominated by a single species or not. To describe the diversity of information, i.e. the uncertainty of the content of information (*entropy*), Claude Shannon developed another famous diversity index — the Shannon index, also called Shannon-Weaver index (Shannon, 1948). Another area in which diversity indexes are used is Economy. The *Gini Coefficient* describes the extent of inequality in

DRAFT

Network	Max.Degree	Centralization	Diversity
Agent x Event	16	0.44	
Agent x Knowledge	4	0.43	
Agent x Location	6	0.55	
Agent x Task	12	0.44	

Table 3.8: Centralization and Diversity of Degree Centrality

a country, e.g. the income distribution (Gini, 1921). The Herfindahl (1945) index (also known as Herfindahl-Hirschman index) measures the concentration of firms in a particular industry. This index is high in case of a single (almost) monopolistic company with very high turnover and some companies with very small turnover. The index tends toward 0 in case of a lot of rather small companies.

All the diversity measures that we have introduced in the previous paragraph have one important characteristic in common, they take a value for every entity that is part of the analysis as input parameters and return a single value for the entire system that describe the distribution of the values from a perspective of equality/inequality perspective. To describe the variance in a network we can use all of them. For now, we use the Hirschman (1945) index. In the last column of Table 3.8 you can find this index applied to the four *Agent* two mode networks.

3.3.3 Correlation

When we hear *correlation* we automatically have Pearson's correlation coefficient (Pearson, 1920) in our minds. But this is just one specific interpretation of correlating entities. In general, correlation measures for networks compare pairs of nodes with each other by looking at their similarities or dissimilarities (Carley, 2002). In the following, we describe four different measures based on the idea of correlation. To keep our examples simple, we are interested in correlation measures for *Agents* based on shared on distinct *Knowledge*. Of course, these measures can be calculated for any network that connects two different node classes, e.g. which *Tasks* need a similar set of *Knowledge*?

For all measures—similarity, distinctiveness, resemblance, and expertise—it is necessary to create a new matrix with the pairwise similarity or dissimilarity in each cell. When you look at these measures in the appendix, you can see that we create these new matrices often via multiplication of a network matrix with itself. What does this mean? Here is an example which is a subset of our Julius Caesar *Agent* x *Knowledge* network. Let us just focus on three people,

DRAFT

Antony, Brutus, and Caesar as well as their *Knowledge*

1. Antony: Citizenry, Military, Persuasion, Politics
2. Brutus: Citizenry, Military, Persuasion, Politics
3. Caesar: Military, Politics

When we now look at each pair of *Agents* and count the number of overlapping *Knowledege* we can create the following matrix. You can find the matrix algebra notation for this transformation on the left side of the matrix:

$$AK \cdot AK^t = \begin{array}{lccc} & \text{Antony} & \text{Brutus} & \text{Caesar} \\ \text{Antony} & 4 & 4 & 2 \\ \text{Brutus} & 4 & 4 & 2 \\ \text{Caesar} & 2 & 2 & 2 \end{array}$$

We do not discuss matrix algebra in this book. In case you are interested in understanding these techniques, look for books like ”Matrix Algebra” in your favorite (online) book store or in particular for ”Matrix Multiplication” in your favorite web search platform. For now, let us focus on the result of the calculation. Every cell in the new matrix represents the number of common *Knowledge* between each pair of *Agents*, e.g. “2” in the Antony / Caesar cell tells us that these two *Agents* share two common *Knowledge* (Military and Politics). The new matrix is, of course, symmetric. The diagonal [1] of the matrix contains the number of *Knowledge* every single *Agent* is connected to, for himself ignoring the *Knowledge* of other *Agents*.

Based on this first step of comparing the similarity of all pairs of *Agents* the following measures are calculated. Some of which need a dissimilarity matrix that can be calculated in a very similar way. Details to the measures can be found in the appendix. The first two measures are node level measures.

Similarity tells us to which extent the other *Agents* have the same *Knowledge* than a specific *Agent*. Based on the similarity matrix that we introduced in the previous paragraphs, the similarity index of an *Agent* is the average of its similarities to all other *Agents*.

Expertise is very similar but focus on the dissimilarity between agents concerning their shared *Knowledge*. Consequently, the inverse *Knowledge* of an *Agent* is compared with the *Knowledge* of all other *Agents*, i.e. the *Knowledge* dissimilarity of Antony and Caesar is 2, while it is 0 for Antony and

[1] The diagonal of a one-mode network matrix holds all cells with the same line and column index, i.e. $A(i, i)$. Therefore, this describes the relationship of a node to itself. This is also called the self-loop.

DRAFT

Agent	Similarity	Expertise
Antony		
Brutus		
Caesar		
⋮		

Table 3.9: Correlation measures of Agents in Caesar's empire

Brutus since they have 100% of their *Knowledge* in common. If we calculate the Similarity and the Expertise index for the main *Agents* of the tragedy, we can see that ... (Table 3.9).

The other two measures of the concept of correlation are dyad level measure. Therefore, the results are not a single value for every node but for every pair of nodes.

Resemblance identifies *Agents* with identical *Knowledge* in a meta-network. Resemblance is maximal in case two *Agents* have exactly the same *Knowledge* and lack in exactly the same *Knowledge*. When we once gain look at the small set of three *Agents* described above, we will find that Brutus and Antony have 100% *Knowledge* resemblance (or 1.0) as they know and don't know exactly the same. Brutus and Caesar share a resemblance score of 0.67 as they both know about Military and Politics, they both don't know about Administration and Prediction, and they have different *Knowledge* about Citizenry and Persuasion (at least in our interpretation of the play).

Distinctiveness. This last part of different *Knowledge* is in the focus of the *Distinctiveness* measure which can be used to calculate the degree to which each pair of *Agents* has complementary *Knowledge*, i.e. *Agents a* knows about something which *Agents b* does not know and vice versa. Distinctiveness is also expressed as the percentage of all available *Knowledge*.

DRAFT

3.3.4 Specialization

The fourth and last concept that we discuss in this section is *Specialization*. Measures of this group try to identify specific *Agents* that have either exclusive or redundant *Knowledge*. Why is it critical to understand *who* specializes in *what* within any network? Within the Julius Caesar Network it would be highly valuable to understand *who* amongst all of his underlings possesses a specialist *Knowledge* relating to the operation of his empire. For the conspirators it would be interesting to know which of the critical *Knowledge* to run the empire is exclusively held by some people or which of their scheduled

Tasks needs a specialized set of *Resources* that are not used for other *Tasks*. Another perspective on Specialization measures is to identify the extent to which the *Knowledge* in a network is redundantly connected to *Agents*. But let us first have a closer look at *Exclusivity*.

Exclusivity in network analysis can be of fundamental interest to discerning the critical infrastructure of a network. Naturally, as the term may allude, we are interested in performing an analysis that will reveal exclusive entities within our network model (Ashworth and Carley, 2006). Consequently, we are looking for any entity within our network model that has an exclusive link to another entity, which occurs across our two-mode network data. In lay parlance, this is finding out who in the network does something or knows something that nobody else does or knows. Such information could be very useful to Caesar in determining his empire's *Organization*al vulnerabilities.

Access takes the idea of Exclusivity one step further. The *Access Index* identifies those *Agents* which are directly or indirectly connected to exclusive *Knowledge*, *Resources*, etc. (Ashworth and Carley, 2006) The result of this measures is a binary value for every *Agent*, i.e. access critical *Agents* get 1 and all other *Agents* get 0. The identification of access critical *Agents* happens in two steps. First, for the *Knowledge Access Index* we identify *Agents* that are exclusively connected to entities of *Knowledge*. These *Agents* are critical. In addition, a second group of *Agents* is critical; those that are exclusively connected to an *Agent* of the first group in the *Agent x Agent* communication or interaction network, because these *Agents* control the social connections to *Agents* with critical access. For instance, in the first act of Julius Caesar the Soothsayer warns Caesar about the "Ides of March". Let us assume the Soothsayer had an inspiration about something bad happening with Caesar in his dreams the night before that event. If we now call this inspiration a piece of *Knowledge* then this *Knowledge* is exclusively connected to the Soothsayer in case that nobody else in the empire had the same dream. Therefore, the Soothsayer has critical access to *Knowledge*. But also Caesar has critical access since the Soothsayer exclusively communicates with Caesar in this matter. The result is obvious, the Soothsayer and Caesar can both block the warning from the other *Agents*.

Redundancy is the risk based on duplication in *Task* assignments, *Resource* access, and *Knowledge* access. For instance, *Knowledge* is redundant if more *Agents* share the same *Knowledge*. Redundancy is a network level measure. Therefore, we get a single value for every two-mode measures for which we are calculating the measure. From the extreme perspective that every available *Knowledge* entity is connected to one *Agent*, Redundancy is

DRAFT

Group	Concept	Description
Quantity	Degree	Count row/column entries in networks
	Load	Average link values/counts of networks
Coherence	Congruence	Proportion of correctly assigned connections
	Needs	Undersupply of assignment
	Waste	Oversupply of assignment
	Performance	Percentage of task completion
	Workload	What is used to perform *Tasks*
Substitution	Availability	*Agents* available for certain *Roles*
	Reuse	Utilization of connections to multiple tasks
	Negotiation	Need for communication to perform *Tasks*
Control	Demand	Cognitive effort expended by *Agent*
	Awareness	Cognitive similarity of *Agents*

Table 3.10: Classification of measure concepts for three and more entity classes

calculated by counting the number of other *Agents* that are also connected to every *Knowledge* entity. An *Organization* with little redundancy is more adversely affected by an *Agent* or *Resource* no longer being available such as a ”Cassius” taking his *Knowledge* and getting a new job in some other empire. On the other hand, too much redundancy makes an *Organization* inefficient. For instance, you wouldn’t want everybody in the imperial Senate to learn how to conduct military operations when everyone has other *Tasks* and accountabilities to see too as well.

3.4 Concepts for three and more entity classes

We have talked a lot about different concepts and measures in this chapter. All the measures, that we have described and used so far, just used two node classes at a time for calculating the results of the measures. The connections of entities from two node classes can be described in one network. For the following concepts and measures, we will use more than two node classes to calculate a single result. Therefore, we need more than one network as input for these measures and we call the underlying concepts of these measures *multi-mode concepts*.

The part for multi-mode concept and measures is organized in a similar way than the previous section describing the two mode concepts. We discuss the ideas behind the concepts at this point and you will find the algorithmic details in the appendix at the end of this book. To describe the underlying logic of meta-network measures, we focus on the four main node classes, *Agents*,

DRAFT

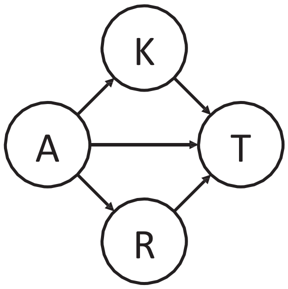


Figure 3.3: Conceptual connections between the four main node classes

Tasks, *Knowledge*, and *Resources*, and how these node classes can be logically connected with each other through networks. Of course, the most important node class are the *Agents*. *Agents* are essential for every Dynamic Network Analysis and are part of almost every measure that we discuss in this book. Beside *Agents*, there is a second node class that is very central for measures that make use of three or even more node classes. When you look back at the enumeration of the ten node classes, which one do you think will be almost as important as *Agents* for the following measures?

The correct answer to this question is: *Tasks*. This may be surprising for you, but Dynamic Network Analysis deals a lot with *Tasks* and the question whether they can be accomplished or not. From the dynamic perspective, *Tasks* are like goals. Actually, developing a product and even killing Caesar are very well defined procedures, all of which having a goal that defines the successful accomplishing of the *Task*. To accomplish *Tasks*, *Agents* are assigned to these *Tasks*. Figure 3.3 shows this connection in the center of the visualization. A circle in this figure represents a node class, while every link stands for an entire network connecting two node classes with each other. *Knowledge* and *Resources* are positioned between *Agents* and *Tasks*. *Agents* need *Resources* and *Knowledge* to accomplish their *Tasks*. Most of the time, the following concepts and measures will discuss this basic system of Dynamic Network Analysis from different perspective.

Coherence measures calculate whether the *Resources* and *Knowledge* of *Agents* that are assigned to *Tasks* actually fit the *Resources* and *Knowledge* that is necessary to accomplish the *Tasks*. Substitution measures look for entities in different node classes whether they can be reused or substituted. Finally, control measures are the most complex DNA measures that are currently available and incorporate different aspects of the connections between the four main node classes in single measures. However, let us start with the

DRAFT

easiest group of measures grouped under the concept of *Quantity*.

3.4.1 Quantity

Quantity measures—similar than for two-mode measures—just count cells in the network matrices.

Degree. A multi-mode degree measure is *Personal Costs*. It counts the number of labor-intensive connections of an *Agent* in the entire meta-network, i.e. the number of people reporting to the *Agent* (in-degree) plus the number of connections to *Knowledge*, *Resources*, *Tasks*, etc (Ashworth and Carley, 2003)

Load measures are similar to the idea of density. We call these measure concept *Load* in meta-networks because they tell us how *loaded* a network is. (Wasserman and Faust, 1994)

DRAFT

3.4.2 Coherence

The biggest group of measures for multi-mode networks is created through the concept of *coherence*. These measures cover the connections of the main node classes as visualized in Figure 3.3. The line in the middle of the picture represents the assignment of *Agents* to *Tasks*. In a nutshell, coherence measures use the networks represented by the other lines of this figure to check whether the *Agents* have the right *Knowledge* and *Resources* to perform their assigned *Tasks* or not. Just as in the previous sections, we focus on *Knowledge* to describe the following measures.

Congruence is a network level measure that calculates the extent to which the *Knowledge* of *Agents* which are assigned to *Tasks* actually fits the *Knowledge* that is actually needed to accomplish the *Tasks* (Carley, 2002). Maximal congruence (1.0) is just achieved in case *Agents* have exactly the *Knowledge* they need for their *Task*—nothing less and nothing more.

Needs is similar to *Knowledge Congruence*, but quantifies only the under supply of *Knowledge* to *Tasks* (Lee and Carley, 2004). *Task Knowledge Needs* compares the *Knowledge* requirements of each *Task* with the *Knowledge* available to the *Task* via *Agents* assigned to it. Needs is a node level measure for *Tasks*. Therefore a value is calculated for every *Task* describing the amount of missing *Knowledge* to accomplish this *Task*.

Waste measures the opposite of *Needs* and is also a node level measure for the *Task* node class (Lee and Carley, 2004). *Agent Knowledge Waste* compares the *Knowledge* of the *Agent* with the *Knowledge* it actually needs to do its *Tasks*. Any unused *Knowledge* is considered wasted.

Performance also calculates a *Task* related measure but results in a network level result. On particular performance measure is *Knowledge Based Task Completion* (Carley, 2002). For this measure the percentage of *Tasks* is determined that cannot be completed because the *Agents* assigned to these *Tasks* lack in needed *Knowledge*.

Negotiation is a network level measures that is based on identifying the proportion of *Tasks* that need negotiation of the *Agents* (Carley, 2002). Negotiation is necessary in case the *Agents* assigned to *Tasks* do not have all the *Knowledge* and *Resources* that are necessary to accomplish the *Tasks*. Based on the congruence of the main node classes *Agents*, *Tasks*, *Resources*, and *Knowledge* the *Tasks* that need negotiation are identified.

3.4.3 Substitution

Availability is a *Role* based measure of *Substitution*. The idea behind *Availability* is that to accomplishing a specific task it often does not need a particular *Agent* but rather an *Agent* fulfilling a certain *Role* (Behrman).

Reuse The Reuse measures (Carley et al., 2000) looks for utilization of *Knowledge* or *Resources*. The question here is whether *Knowledge* that is needed for a *Task* is already available in an Organization/Company because the particular piece of *Knowledge* or *Resource* is already used for other *Tasks*. Reuse measures are often referred to *Omega* measures.

3.4.4 Control

Control measures express the extent to which agents are in control about the entire meta-network or about the situation of the other *Agents* of the network. Both measures of this concept can be referred to as *Cognitive* measures.

Cognitive Demand is a node level measures that incorporates ten different aspects of *Agents* being cognitively engaged in a meta-network (Carley, 2002). Some of these aspects can just be calculated in case you have *Knowledge* and *Resources* in your data. In case your meta-network includes just one of these node classes as well as *Agents* and *Tasks*, you can still compute *Cognitive Demand* by just calculating the feasible aspects. The ten aspects covered by *Cognitive Demand* are the following:

1. The number of other *Agents* that a single *Agent* is connected to.

DRAFT

2. The number of *Tasks* an *Agent* is assigned to.
3. The amount of *Knowledge* an *Agent* has.
4. The amount of *Resources* an *Agent* is connected to.
5. The number of *Agents* that are assigned to the same *Tasks* than the *Agent*.
6. The amount of *Knowledge* that is necessary to accomplish the *Tasks* that an *Agent* is assigned to.
7. The amount of *Resources* that is necessary to accomplish the *Tasks* that an *Agent* is assigned to.
8. The amount of negotiation on *Resources* with other *Agents* that is necessary to accomplish the assigned *Tasks*.
9. The amount of negotiation on *Knowledge* with other *Agents* that is necessary to accomplish the assigned *Tasks*.
10. The number of *Agents* that an *Agent* depends on or that depend on an *Agent* to handle his *Task*.

This enumeration of different aspects that are incorporated in *Cognitive Demand* makes it clear that people that score high in this measure are very important in a network. We call these people "*Emergent Leaders*". Emergent leaders are identified in terms of the amount of cognitive effort that is inferred to be expended based on the individual's position in the meta-network. Individuals who are strong emergent leaders are likely to be not just connected to many people, organizations, tasks, events, areas of expertise, and resources; but also, are engaged in complex tasks where they may not have all the needed resources or knowledge and so have to coordinate with others, or have other reasons why they need to coordinate or share data or resources. The drawback for Emergent Leaders is that they are so busy with keeping the company (or the empire) running, that there is a good chance that they never become a formal leader.

Awareness is another control measure that creates dyadic results (Graham et al., 2004). The measure tells us to which extent two *Agents* are similar from a meta-network perspective. *Shared Situation Awareness* is a mixture of four different aspects:

1. The interaction between the two *Agents*.

DRAFT

2. The power of the two *Agents* calculated by their *Eigenvector Centrality*.
3. The *Agents* physical proximity approximated through their shared *Locations* or *Events*.
4. The *Agents* socio-demographic similarity approximated through their shared *Knowledge* and *Resources*.

This enumeration covers different aspects of having shared awareness about the networked system.

3.5 Problem set

1. What are the main four nodes classes of dynamic network analysis?
2. Which two node classes occur in almost every measure that uses three or more node classes?
3. What is the difference between the quantity concept of two-mode measures and the quantity concept of multi-mode measures?
4. Describe the underlying idea of coherence measures.
5. Why are people critical in a network that score high in *Exclusivity*?
6. Which situations create the need for *Negotiation* in a company?
7. Draw a small network consisting of *Agents* and *Resources* that describes the *Access* measure.
8. What do you think, is it likely that the person in an *Organization* that scores highest in *Cognitive Demand* will become the next leader of this *Organization*?
9. In Figure **??** you can find the conceptual linkage between *Agents*, *Tasks*, *Knowledge*, and *Resources*. Find another combination of four node classes that can be used for any coherence measure.
10. What is the main difference between *Congruence*, on the one hand, and *Needs* and *Waste*, on the other hand?
11. *Imagine that you are the head of a company and you measure high *Redundancy* in one department. Why could this be a problem? And why could this be important?

DRAFT

12. *Imagine that you had to calculate betweenness centrality in the network that is an aggregated representation of all five acts. Why is the generic node *Citizens* problematic for the centrality score of ALL nodes of the network?

13. *What is the 11th node class and why could analysis involving this node class make sense?

14. *Calculate the *Coherence* measures for the Julius Caesar meta-network and give an interpretation of the results.

15. *Download the company network from the book's website. Identify the cognitive leader of the company.

16. *The *Event* x *Event* network in Figure 3.2 consists of long chains of events. Why is this the case? Is this typical for real world *Event* networks?

17. **Take a sheet of paper. Copy the *Agent* x *Knowledge* network matrix and perform a matrix multiplication by hand so that the result is a *Knowledge* x *Knowledge* matrix. What is the meaning of the resulting matrix?

18. **Calculate by hand the network level measure "negotiation". Describe the different steps of the calculation.

19. **Take your favorite movie and code as many node classes as possible. What nodes classes are tricky to code? Visualize and analyze this meta-network. Can you describe the narrative as well as the highlights of the movie with network visualizations and metrics? There is no need to create more than one meta-network for the entire movie.

20. **Take all two-mode and multi-mode measure concepts and find a new classification system that does not use the number of entity classes as first separator.

DRAFT

DRAFT

Chapter 4

Finding Groups

DRAFT

Imagine, Julius Caesar is going through some major organizational restructuring. The soldiers are naturally worried, but the whole idea is for the empire to realign strategically to allow the groups within the empire the most efficient use of Resources. It is the theory of Julius Caesar that the empire can then become more profitable and hence more stable. Therefore, ultimately, it is about reassuring the soldiers of the empire that Julius Caesar will continue to remain strong long into the near future. Now, that should make some rest easier but perhaps it makes politicians, especially Cassius, a little uneasy. After all, they have given themselves a daunting task: they want to know about the groups that exist inside their organizational structure. They believe, quite correctly, that knowing the make-up of how groups and teams function across the empire will therein provide clues as to where Julius Caesars empire can reallocate Resources to make Julius Caesars empire more efficient and hence more enduring. The problem is how do they go about discovering the groups that exist within their empire?

Inside Julius Caesars empire, we have Cassius, who interacts with Brutus and Calpurnia. Everyone interacts within his or her group. We will say that they are part of distinct organizational group called Roman Countrymen but could there be other groups they interact with outside of their core group? Of course there could. There could be countless amounts of other groups which they interact with on a regular basis. Cassius could hang out with the accounting group while on lunch. He may have privy to Knowledge that is generally available only to those in the friends group. Likewise, Brutus might take Roman baths on the weekends and belong to the same club that many of the senior leaders belong too. He may therefore have the inside scoop on a lot of high-level strategy that generally would be unavailable to anyone within the customer service group. He may even serve as a source of information to the senior executives as to what really works in the custumer service group.

Groups are a natural part of our human experience. Individuals are indeed social actors who tend to migrate into membership into a group, or several groups, for a variety of reasons. In prehistory ages, the human species seemed to have figured out that the family group is the starting social structure for our very core survival and that being part of an even larger group provides us with increased access to critical resources, such as food, water, potential mates, etc. Moreover, being part of a group afforded the individual better protection from our enemies, both the difficulties of nature and the brutalities of other humans. Today, this drive for affiliation continues; we each are a member of numerous groups, from personal-contact groups, to cyber-space virtual world groups.

From a DNA perspective, groups also encapsulate objects and ideas, such as computer networks, music genres, and job categories. Whether by the force of a natural law, or the limited capacity of humans, objects and ideas are placed into groups. Groups of all types, forms and made up of various complements and, even, subgroups are everywhere.

So it follows, that a network analysis will typically investigate a network data from the perspective of the network as a group or a set of groups. This perspective accomplishes two things: (1) it reduces the number of entities to analyze, and (2) the natural tendency for things (including people) to gravitate into groups, suggests that understanding the sub-groups in a network may provide hidden clues to that network. So, people are interested in groups is, because when you look at groups in a network, it is a very easy way to summarize information in a network. Rather than saying something about any particular node, we are trying to aggregate information to higher representations groups. This is especially helpful when the network is large. When we think of groups we automatically think of communities that share knowledge, information, or norms or have similar experiences. So, you could interact with them in similar ways and you can expect similar reactions from members of the same group.

In this chapter we will first introduce some interesting patters in networks which can be considered as groups, e.g. disconnected components or sub-sets of nodes in which every node is connected to every other node. In the second part of this chapter we will discuss more complicated group structures—those which need algorithms to be identified. Finally, we will have a closer look at our Julius Caesar network to discuss grouping issues in meta-networks or in

DRAFT

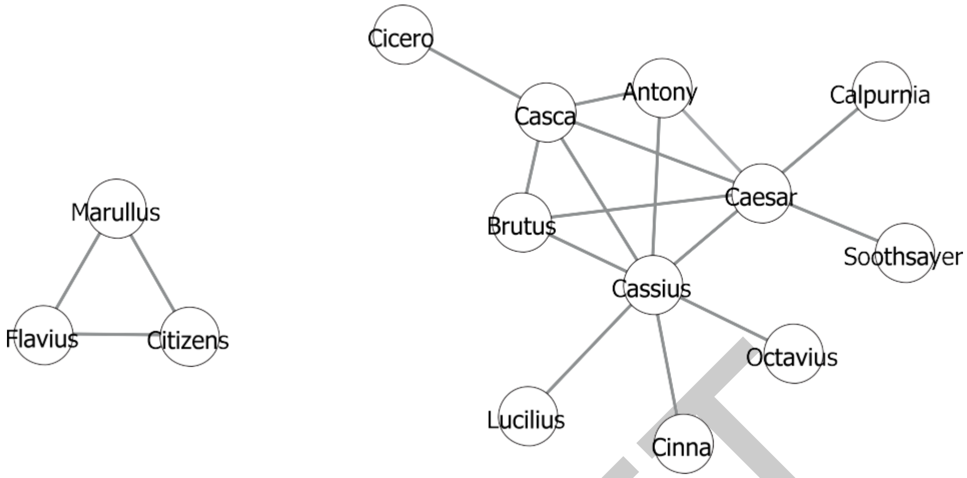


Figure 4.1: Network visualization of the first act of Julius Caesar

DRAFT

networks over time.

In general, groups are a set of nodes that meet certain criteria. This sounds easy but creates a major implication: There are no *natural* groups in networks; you create groups based on your definition. Therefore, there is no one-fits-all method to identify groups. This is the reason why many new grouping algorithms are being developed every year and grouping is a big topic in DNA. When working with groups in the network setting, there is not yet agreement on a precise definition (Seidman & Foster 1978; Alba & Moore 1978; Mokken 1979; Burt 1980; Freeman 1984; Freeman 1992; Sailer & Gaulin 1984). Exactly what constitutes a group is a matter of ones perspective, the data, and analytic question. For our purposes here: cohesive sub-groups, or just groups, are collections of nodes that share some specific characteristic(s) or property(ies); most often a nodes membership is bounded by being in only one sub-group, but some grouping processes allow for multiple memberships. We seek to identify and characterize these groups because often the analyst can locate, often hidden, sets of actors that have something (loosely defined) in common that can be useful information in explaining, understanding and forecasting, either the group itself or the larger network. As mentioned earlier, these often-hidden groups are sometimes (self-) organized for some collective action of sorts.

Let's have a look at Act 1 of the Julius Caesar play again (Figure 4.1. What *Agents* form a group? Well, as you would probably know by now, this depends on the selected criteria.

The first criteria that we look at is not directly connected to networks. *Agents* share different *attributes* and form groups based on that. For instance, age, gender, occupation, or sympathy to Caesar can be used to group the *Agents* of reffig:act1. Sharing specific attributes can lead to *nominal groups*, e.g. nurses, teachers, or senators. Even though attributes include no network information at first sight, we know from chapter 2 that similarity in attributes (homophily) is number one drive for network formation. Conspirators tend to hang out with other conspirators and new *Agents* are more likely of being accepted by the group when they share this attribute.

4.1 Interesting patterns

A second set of grouping criteria are based on *graph theoretic* definition. When we look at the structure of the network, the most obvious *groups* in this network are the two unconnected parts, Marullus and Flavius together with the Citizens of the left side and all other nodes on the right side. We call these unconnected parts of a network *components*. Are these different scenes of the play? No, components are created when there is no path between (at least) two groups on nodes. For the play this means, when there are two *Agents* from different scene that do not co-occur with any *Agent* that can create a chain of links between them.

Another interesting structural pattern is created by Antony, Cascar, Brutus, Cassius, and Caesar. When there is a group of nodes with each of which is connected to every other node, then we call it a *clique*. Cliques are the strongest pattern of structural pattern but you need to be careful using cliques, when your network data is created by folding two-mode data to one-mode data, as we did for the network in reffig:act1. Every node of the second mode forms a clique in the folded one-mode network. This can create confusing artifacts of large cliques in case of, let us say *Events*, with many participants.

Cliques are very intriguing, however, not often used in real-world networks. There are two reasons for why this group criteria is not so great after all. First, in real-world groups that exist of a certain number of *Agents*, some links are often missing even in the strongest group of friends or co-workers. Secondly, cliques are very time consuming to calculate for larger networks.

Some other concepts for group identification are similar to cliques but are more *realistic* and very fast to calculate. *k-Cores* **??** are groups of nodes that are all connected to at least *k* other nodes of the groups. Therefore, we can use this approach to identify *clique-like* groups. *k-Plex* **??** is a similar concept. Here, a group is defined as a set of nodes that ... ??. It is worthwhile knowing

DRAFT

that *k-cores* and *k-plexes* as well as *cliques* can and often do overlap—one node can be part of more than one group.

Equivalence i Equivalence Structurally indistinguishable Same degree, centrality, belong to same number of cliques, etc. Only the label on the node can distinguish it from those equivalent to it. Perfectly substitutable: same contacts, resources

Face the same social environment Similar forces affecting them same influencers On average, hear things equally early, influenced similarly, have similar things to cope with

Role equivalent

4.2 Grouping Methods

When groups are *extracted* from networks with grouping algorithms, they are often called *communities*. A community consists of a subset of nodes within which the node-node

4.2.1 Newman grouping

connections are dense, and the edges to nodes in other communities are less dense. Be aware that this is just another criteria definition for groups—although one that is highly relevant for networks consisting of people. The grouping algorithm by **?** is very often used to identify communities because it is a very intuitive approach. Remember betweenness centrality (Freeman, 1977). We discussed in chapter 2 that a node scores high in this measure in case this node links parts of the network that were not connected as well without this node. It is clear that the nodes 1, 2, and 3 in the network visualized in 4.2 fulfill this criteria while peripheral nodes have a betweenness centrality score of 0.0. Now the same idea is used for links of the network. The link with the highest betweenness centrality is considered to connecting different groups that were not connected as well without this link. In Figure 4.2 the line width are drawn based on line betweenness centrality. You can see that the connecting line between node 1 and 2 scores highest. The smart algorithmic move now is, to remove this link. By doing so, the example network breaks in two components. Of course, in real networks network parts are not connected by just one link. Therefore, we recalculate line betweenness centrality again, delete the line that scores highest, etc. Recalculating the metrics again after removing a link is necessary as the the results may change

DRAFT

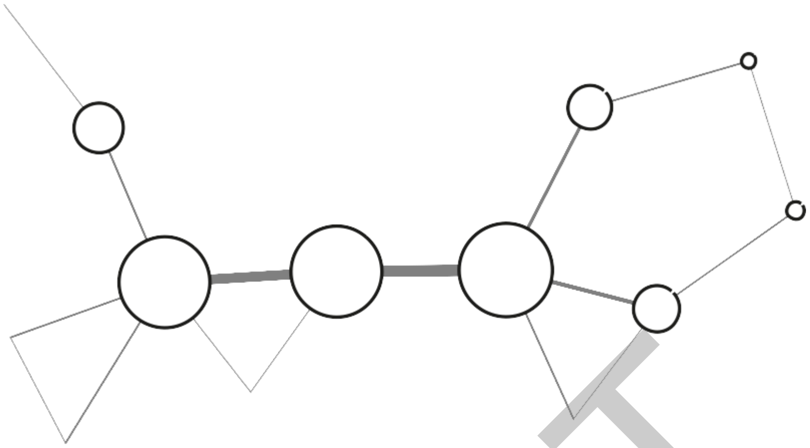


Figure 4.2: Example for line betweenness centrality

dramatically. In our example, the link between 1 and 3 would looses almost all of its centrality after the link between 1 and 2 is removed. The removing of links can be calculated till no link is left. To answer the question when to stop, **?** offered the *modularity* measure that xxx

The disadvantage of this approach is obvious when you recall the procedure. Calculating betweenness centrality again and again is costly and impossible for larger networks.

Newman [2006] : Start inside community and search for boundary. Relatively fast for large networks

4.2.2 CONCOR grouping

CONCOR puts its focus on the notion of structural equivalence to discriminate among the nodes to form groups. Structural equivalence is the notion that two nodes that have the same number of links with the same alters are therefore structurally equivalent. CONCOR essentially performs often multiple, row or column-wise, vector correlations to determine the level of structural equivalence between a given pair of nodes. This correlation process is repeated until ultimately the matrix representing the network has stabilized with a set of 0s and 1s. These values indicate to which group an individual node will be placed. Notice, this situates CONCOR to creating only up to two groups. CONCOR can repeat itself multiple times to further split one or both of the previously located groups. Therefore, CONCOR most often produces

DRAFT

a number of groups that are a power of two. The number of groups identified by CONCOR is a user-parameter (often times the number of "splits" is the expected input). PRO: Only commonly used algorithm detecting relaxed structural equivalence. CON: Top down splitting of nodes imposes structure CON: Requires user to choose a power of 2 for the number of groups.

4.2.3 Johnson grouping

The Johnson procedure uses a distance metric to discriminate groups. It creates groups according to a network that is constructed with links that indicate the distance between dyads. Johnson will segregate groups according to these weights by separating those who are most distant from others in the same group from the original group and off to a group of nodes that are closer in this distance value.

4.2.4 Fuzzy Grouping

The FOG / K-FOG are powerful approaches that recognize that nodes can often be members of more than one group at a time (fuzzy groups), which can be a major weakness of the aforementioned techniques. FOG begins with a collection of nodes and uses a maximum likelihood perspective to determine the probability that a link exists among the various dyads in the group. It compares the probability with the actual data and as a result, a node can indeed be assigned membership in the group, or not. By taking this approach, FOG can assign a node to multiple groups.

DRAFT

4.2.5 Block-Modeling

Block-Modeling **??** is a grouping approach that optimizes the matrix representation of a network. A block model is a reduced form representation such that nodes are divided into a set of mutually exclusive groups. The resulting groups can then be analyzed as a network such that

The groups connection to itself is the density of the connections among members

For each pair of groups, the inter-group connection is the density of the connections of group 1 (row) to group 2 (column)

The resulting block matrix can be turned into a binary matrix by simply comparing the level of connections in the block to the overall density of the original matrix such that there if the value of the cell is ¿= to the overall density then we replace it with a 1, else 0

		1	2	3	4	5	6	7	8	9	10	11	12	13	14
Antony	1	.	.	1	1	1	.	.	.	.	.	.	.	.	.
Brutus	2	.	.	1	1	1	.	.	.	.	.	.	.	.	.
Caesar	3	1	1	.	1	1	1	.	.	.	.	1	.	.	.
Casca	5	1	1	1	.	1	.	1	.	.	.	.	.	.	.
Cassius	6	1	1	1	1	.	.	.	1	1	1	.	.	.	.
Calpurnia	4	.	.	1	.	.	.	.	.	.	.	.	.	.	.
Cicero	7	.	.	.	1	.	.	.	.	.	.	.	.	.	.
Cinna	8	.	.	.	.	1	.	.	.	.	.	.	.	.	.
Lucilius	11	.	.	.	.	1	.	.	.	.	.	.	.	.	.
Octavius	13	.	.	.	.	1	.	.	.	.	.	.	.	.	.
Soothsayer	14	.	.	1	.	.	.	.	.	.	.	.	.	.	.
Marullus	12	.	.	.	.	.	.	.	.	.	.	.	.	1	1
Flavius	10	.	.	.	.	.	.	.	.	.	.	.	1	.	1
Citizens	9	.	.	.	.	.	.	.	.	.	.	.	1	1	.

Table 4.1: Block modeling the social network matrix of the first act of Julius Caesar

In Table 2.2 on page ?? the social network matrix of the first act of Julius Caesar was shown. Block-modeling makes use of the fact that changing the order or the lines (and to the same extend the order of the columns) does not change the network. Therefore, we can rearrange the lines and columns for whatever purpose.

Opposing groups 10-01
Hierarchy 10-10
Core-periphery 11-10

4.2.6 Other approaches

Methods from multivariate statistics.
Single linkage; connectedness; minimum
Complete linkage; diameter; maximum
k-means

4.3 Groups in meta-networks

4.4 Problem set

DRAFT

1.

2.

3.

4.

5.

6.

7.

8.

9.

10.

11. *

12. *

13. *

14. *

15. *

16. *

17. **

18. **

19. **

20. **

DRAFT

DRAFT

Chapter 5

Spatially Embedded Networks

DRAFT

In exploring the various influences and roles in human relationships we will now consider one of the most important features for understanding social interactions: space. Clearly, space is important, and as such, we intend to equip the reader with appropriate methods for analyzing networks where locational information is available. Physical proximity has long been known to play a major role in shaping human interpersonal relationships. In general, people are more likely to interact with others who are nearby. This effect, called propinquity, has been documented time and time again for a wide variety of networks (Butts, 2002; Faust, 2000; Festinger, 1950; Latane, 1995). It has even been proven theoretically that for large social networks, the locations in space of individuals can explain almost all of the information in the network (Butts, 2002).

This chapter should provide an overview of the core issues in analyzing spatially embedded networks. After discussing different aspects of networks and space—including the importance of aggregation, clustering, information loss and smoothing—we introduce technical artifacts that are needed to handle geographical information. Section 3 continues with spatial visualizations and the final section discusses centrality measures that make use of spacial information.

5.1 Propinquity – Those close by form a tie

5.2 GIS, shape-files, and Co.

5.3 Spatial visualizations

5.4 Spatial centralities

DRAFT

Chapter 6

Temporal Networks

DRAFT

We should know by now that networks evolve and change over time and it is the key role of dynamic network analysis to identify and describe those changes. Furthermore, by analyzing change in networks, we are maybe capable of predicting how networks evolve in the future and how the underlying real-world system will change. In fact, it is this time consideration that marks the true difference between Dynamic Network Analysis (DNA) and traditional link analysis.

In this chapter we discuss different aspects of temporal networks. After introducing some definitions and discussing aggregation issues, we try to describe and measure change in networks. You will learn in this chapter that the statistical analysis of correlating different networks to identify similarities of multiple networks is possible, but not trivial. The final part of this chapter introduces different ways to detect change and periodicities in networks over time. Some of the measures that we discuss in this chapter are mathematically challenging. We try our best to describe the underlying ideas of the methods in an understandable way here and we want to refer the advanced reader to the algorithms in the appendix or the cited literature.

6.1 Networks over time

6.1.1 Creating networks over time

When network analysts talk about temporal networks, they normally talk about networks that are created through data aggregation for a specific time period (e.g. by day, week, month, year). For instance, if you think about e-mail communication in a company, then all e-mail that are sent at one day can be grouped together to form the communication network for this particular day. Another example is our Julius Caesar data. When coding the social interaction of this network (see chapter 2), we decided to aggregate on act level. This was a deliberate decision. Both aggregation levels—for the Julius Caesar as well as for the e-mail network— also can be selected at a different level. We talk more about aggregation on network data later in this chapter.

In general, there are two different ways of describing change in meta-networks:

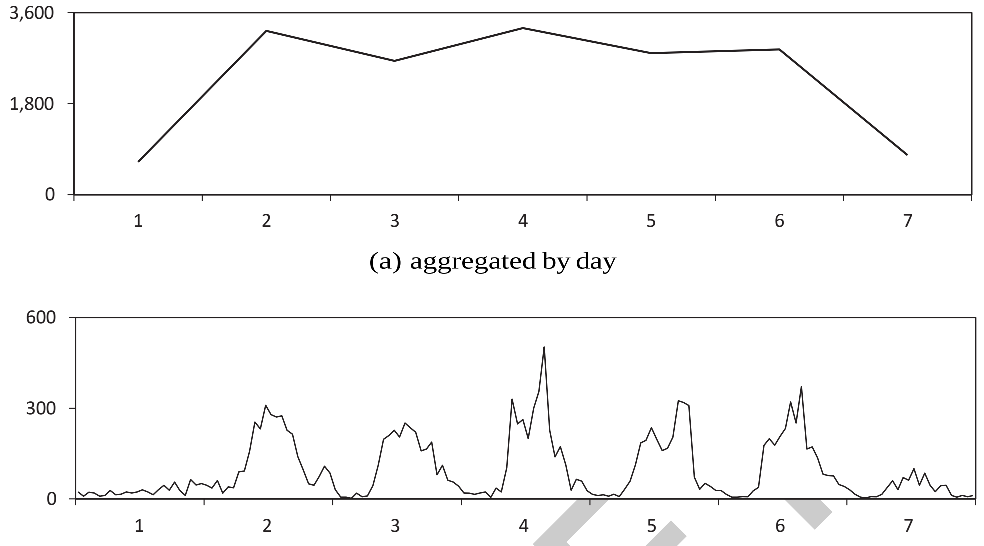
1. *Keyframes*. A meta-network over time is collected as a set of networks, e.g. one meta-network per day, month, year.
2. *Deltas*. A meta-network as a set of single "change events" in time. These change events can be: add/remove a node, add/remove a link, and modify an attribute.

As we discussed above, keyframes are the common way of describing change in networks. But keyframes and deltas are highly connected with each other. Remember the Julius Caesar social network from chapter 2. The act-by-act networks are not just an aggregation by scene, actually theses networks are the result of aggregating every single interaction in every scene, e.g. Calpurnia (the wife of Caesar) tries to convince Caesar not to go to the Senate and Decius does the opposite; both interactions (deltas) are coded in the second act of our network as one edge. Imagine that we have coded every single interaction as delta, we could create the act-by-act keyframes easily by aggregating several deltas within the specified period of of the play. However, we will discuss temporal networks in this chapter from the perspective of keyframes.

6.1.2 Levels of aggregation

We already used the term *aggregation* in this chapter. Aggregating over time data means creating keyframes. To illustrate this process and to discuss issues of aggregation, we look at a dataset of e-mail communication. The dataset includes approximately 200,000 e-mails over the course of 81 days. These large amount of e-mails were sent from 2,427 people to 2,563 recipients. Let us just look at one week of this dataset including 16,277 e-mails. If we start to think about how to analyze this data with statistical methods, a very intuitive approach is to count the number of e-mails by day or by hour. Figure 6.1 shows the result of this statistical approach. Figure 6.1(a) reveals very little activity on Saturday (day 7) and Sunday (day 1) and rather stable activity during

DRAFT



(a) aggregated by day

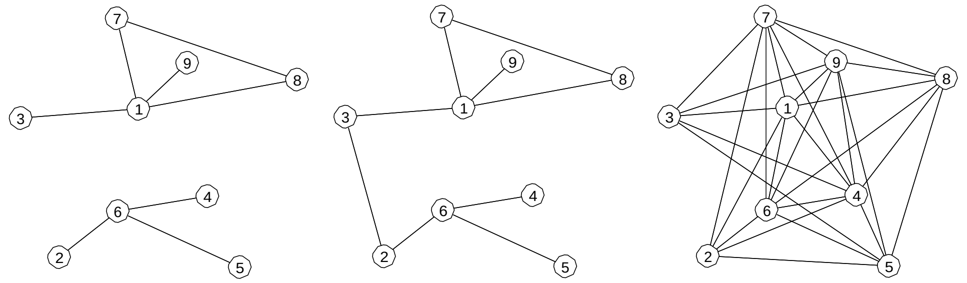
(b) aggregated by hour

Figure 6.1: Number of e-mails aggregated by day and by hour.

the working week. The picture looks different for an hourly aggregation level (Figure 6.1(b)). Here it seems that Wednesday creates much higher activity but looking more closely we can also identify the Monday night activities that led to the high activity level on Monday in the first chart.

So, both charts tell a different story. But why is this connected to networks? From a network perspective, every single e-mail represents a link in the network. And every single link can change a network. Therefore, different aggregation level create different networks! Imagine a person in this organization that is part-time worker and just works in the morning. On a daily aggregation we would see an average active person even if this person sends a lot of e-mails per hour. Another example is illustrated in Figure 6.2. This is a network created from an artificially created company but this data could also be a subset of the e-mail data from Figure 6.1. If we look at the first picture and analyze the company network by hour, we would identify two sub-groups of communication with nodes number *1* and *6* as central in these groups. On a daily aggregated analysis (Figure 6.2(b)) we identify the additional connection between nodes *3* and *2* because these two employees collaborate in a project and had a project meeting in the afternoon. This connection changes the network structure in a way that from a perspective of communication flow (e.g., betweenness centrality) these two nodes are now very important. The

DRAFT



(a) aggregation by hour

(b) aggregation by day

(c) aggregation by week

Figure 6.2: Networks created based on three different aggregation level.

DRAFT

weekly observation, again, looks different because now we identify that there is—for some reason—a lot of communication in this company between almost all people and especially node number *4* seems to communicate with all other nodes. So, what is the the right aggregation level? It depends on your research question and your data. If it is possible, try different aggregation levels analyze (and interpret) the differences. If your aggregation level is selected *wrong*, these are possible implications:

- **Aggregation level too high.** The network gets dense and is harder to analyze. Interesting structures are masked by more or less *random* interactions. It is also possible that two or more interesting pattern superimpose each other.
- **Aggregation level too low.** Miss of important structural patterns. Overemphasize of links having high communication. Very sparse networks result in highly fluctuating measures as a single link can change the global structure of the network.

6.2 Trails

In the last chapter we discussed ways of analyzing network data when it is connected to geo-spatial *Locations*. Now, let us add the perspective of time to this data. For instance, Julius Caesar has seen his empire and bureaucracy ever expanding. He analyzed who the important *Agents* in his empire are and he already accomplished some spatial analysis regarding his empire. In an effort to better analyze the structure of his empire, Caesar decides that he wants to know exactly how his military command travels the known world.

Could the emerging travel patterns lead Caesar and his staff to draw new conclusions about where the business is located or where it should open new *Locations* to better serve the geographic needs of the empire? And what about all the *Locations* that are shared in common by Caesar's military and administration stuff, such as cities they passed through, houses they stayed at in various cities, routes they took, stops they made, on their way to forming the enemy army? In other words, how can we analyze network data in space and time?

When observed over time, spatially embedded networks exhibit a specific kind of dynamism deserving of its own forms of analysis. *Agents* occupy only one *Location* at a time, but progress from *Location* to *Location* longitudinally, creating a temporally embedded sequence of relationships we call a ”trail”. Trails are paths that *whos* move through within a network. Naturally, trails involve both a *who* and *where* and even a *when*. When you can link those entities together over period you have a trail.

Trails are just one perspective on one part of the larger dynamic network, but thinking about relationships in sequence makes certain kinds of analysis much more intuitive. Using trails, we can begin to analyze questions like, ”Where do people at *Location* X tend to go next?”, ”Which other *Agents* does *Agent* A frequently cross paths with?”, or ”What kind of seasonal patterns govern movements in my networks?”

Stepping back from the spatial context, we can see that sequential relationships, and the type of question we ask about them above, are not limited to tracking movement. We might also consider changes in *Agent* affiliation, such as an *Agent* x Employer relationship, or changes of power, such as a Country x Political Party relationship. The formal, generalized definition of ”trail” is (1) a subject node (such as an *Agent*) and (2) a time-labeled sequence of target nodes from the same class (such as *Locations*). Any dynamic relation can be used as a trail set, so long as it has the property that at any given time—it is many-to-one (an *Agent* can occupy only one *Location* but a *Location* may host many *Agents*).

Having established that general view, we will return to discussing trails in their most intuitive context, as a description of spatial transitions. Since trails and networks are closely related and defined in this way, trails are actually a type of network. Although analyzing the trail as a network may not seem interesting, trails can be used to create useful networks. For example, trails can be used to create co-*Location* or co-affiliation networks, showing who was at the same place or *Organization* at the same time. Trails can also be used to create transition networks showing how people in aggregate tend to move

DRAFT

from place to place or from *Organization* to *Organization* (2008b). Trails can also be generated from networks. Although networks do not have sufficient information to reproduce trails, networks can be used to create prototypical trails that might be expected given e.g. a transition network (Davis, 2008).

So, when we look for trails in networks over time, we look for interesting patterns and shared similar behaviors.

ToDo: example of trails, e.g. Davis, 2008. Focus: What can be analyzed?

DRAFT

6.3 Measuring change

Questions for Comparison: Are two networks similar? What is the difference of two networks? How to compare more than two networks? How to compare predicted networks to the actual future observed networks? Can we use standard statistics (e.g. correlations)?

6.3.1 Levels of comparison

We distinguish between four levels of comparison based on what is calculated for every network:

Node measures. Measures are calculated on node level (e.g. centrality measures) and the results are compared with each other, e.g. comparison of the set to top 10 key entities, the ranks of the entities, or the centrality scores of the entities. We did this in chapter 2 with the *Agents* in the five acts of the Julius Caesar network.

Network measures. Network level indices are calculated (distribution of node measure values, network centralization, density, etc.) and then compared between different networks. This approach is similar to comparing node measures.

Network structure. The network matrices are compared with each other, e.g. Hamming distance, Euclidean distances, correlation of networks, regression models with networks. The remaining sections of this chapter will primarily deal with these topics.

Motifs. Local patterns in networks are analyzed and compared between networks to describe the dynamics of change, e.g. transitivity, reciprocity. One approach to compare motifs is the *triad census* from Holland and Leinhardt

(1976). A more comprehensive method is called *Exponential Random Graph* modeling and is presented later in this chapter.

6.3.2 Network distances

One straightforward approach to compare networks with each other to identify the amount of change over time is to calculate the difference between networks. So exactly, how does the network scientist measure the differences between two networks? Well, one the short answer is that they dynamic network analyst is interested in learning about the distance between the network. To do so, is to consider the network matrix as list of numbers. The most common means for doing so is to calculate the *Hamming distance* between two networks. Another method is to use Euclidean geometry to ascertain differences. Of course, there are other methods, each of which is done differently and warrants its own respective considered by the analysts.

Hamming distance. Hamming distance considers network matrices as strings of binary information. Therefore, every cell of the string is either 1 or 0. In Figure 6.3 we illustrate the process of comparing two networks by calculating the hamming distance. You can see visualization and matrix of two different networks, each consisting of 5 nodes and directed links connecting the nodes. On the lower left part of the Figure, the two network matrices are transformed into a string by concatenating the lines of the matrices. If we now compare these two strings bit-by-bit (column-by-column), we can count the number of disparity—in our example 5 elements are different resulting in a proportion of 20 % difference. Consequently the Hamming distance between our two example networks is 0.2.

Let us look again to the Tragedy of Julius Caesar. When we look at the social networks that we have introduced in the first chapter and we want to calculate the hamming distance between the first and the second network, the second and the thirds network, and so on, we are facing a problem—the networks are of different size and when we concatenate the lines of the matrices the resulting strings are also of different size which makes it impossible to compare them. In addition, we can find *Agents* in the second network that are not part of the first and vice versa. And this is not the exemption but rather the rule because in real world—and in written tragedies—different networks barely ever have the exact same *Agents*. To overcome this constraint that we have two options adjust networks so that they have the same size before calculating Hamming distance. Both methods result in different results and you have to be aware of what you are interested in:

DRAFT

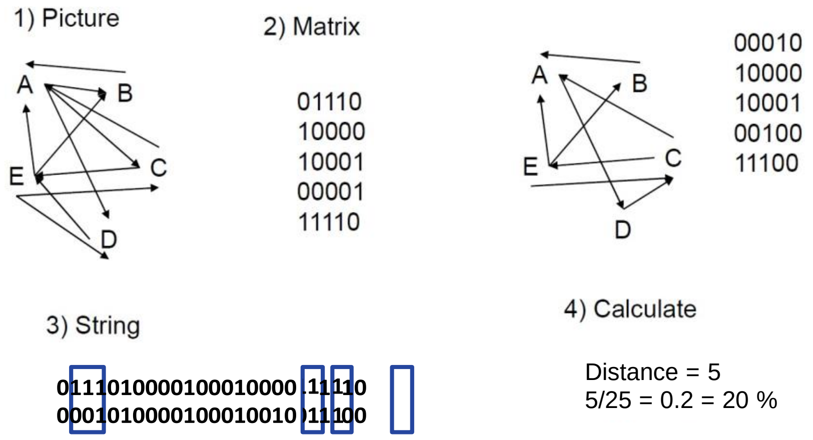


Figure 6.3: Hamming distance of two networks

- *Union*. Add missing nodes to both networks so that the networks consists of all nodes that occur in at least one network. The added nodes have no links.
- *Interesect*. Delete all nodes that just occur in one network.

In a nutshell, the Hamming distance is defined for binary networks as the sum of differences between two networks. In other words, how many changes from 0 to 1 or from 1 to 0 are necessary? This is a very simple and intuitive approach to describe differences between networks.

Euclidean distance. Do you remember the *Pythagorean theorem* from your math class in high school? $a^2 + b^2 = c^2$ defines the relation between the two shorter and the longer line of a triangle. Imagine the Cartesian coordinates (the cross where the horizontal axis is called x and the vertical y). If you start in the origin at (0, 0) (where the axes cross) and move 3 units to the right and from there 4 units to the top. What is the distance from this point to the origin? Correct, it is 5 because based on the Pythagorean theorem, the long side of the triangle is the square root of the sum of the squares of the other two lines, or to say it in math: $\sqrt{3^2 + 4^2} = 5$. This distance of the longer line of the triangle is actually the Euclidean distance from the origin on a 2-dimensional surface. In general, we can use the same concept to describe the distance in any higher dimension:

$$d(A, B) = \sqrt{(A_{11} - B_{11})^2 + (A_{12} - B_{12})^2 + \cdots + (A_{nm} - B_{nm})^2}$$

So, even if we are not capable of imagining a 25 dimensional space, it is quite

DRAFT

easy to take 25 times each with two numbers, calculate the difference, square the difference, [1]. sum up all sub results and calculate the square root from the sum. If you do so and calculate the Euclidean distance for the example of 6.3 you will probably figure out that the result is the square root of the absolute Hamming distance. This is the case because our network is binary and the square of 1 (or -1) is also 1. However, Euclidean distance has a very important advantage compared to Hamming distance, it takes link weights into account. Therefore, if your network is weighted and you are interested in distances between networks, than Euclidean distance is probably your preferred choice.

6.3.3 Correlation of networks and its problems

Another approach to compare networks is to correlate its network matrices. Person correlation (Pearson, 1920) compares any list of numbers and tells us how similar these numbers are. People run their statistical tools and calculate correlations because they are interested in comparing different variables with each other. For instance, managers are interested in whether employees with higher income have higher productivity, teacher want to know whether middle school students that perform better in math do also better in physics, and Caesar could be interested in whether the happiness of his subjects is higher in parts of his empire in which more gladiator fights are happening or in which the taxes are lower. All these examples have in common that different attributes of set of people are compared with each other. When we compare networks we are interested in correlations of relations. To modify the previous examples to relational questions, managers are interested in whether employees that share more common projects interact more often which each other, teachers can be interested in whether students use their friendship ties when they ask each other for advice, and Caesar could be interested in whether private relations (e.g. marriages between families) or business relations between the senators result in a common voting behavior in the Senate. The naïve approach to these network correlation questions is *almost* right.

If we transform the network matrices into lists of numbers (like for calculating distances), we can correlate these lists. For the two networks used in the distance examples, the Pearson correlation coefficient is 0.53 if we ignore the diagonal elements. At first glance, to Calculate the correlation between the edge values in two networks, seems to be a straightforward–and also quite intuitive— approach to compare network matrices. But after careful con-

[1] Squaring the differences has the big advantage that the ordering of the two numbers is not relevant for the results. In case the difference is negative, the square is also positive

DRAFT

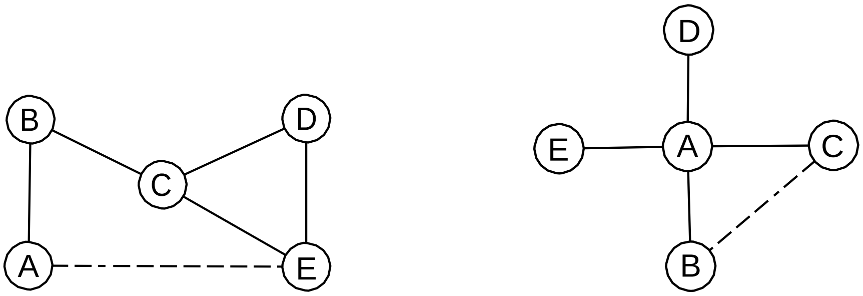
sideration we can identify some major problems concerning correlation on network matrices because network data violate basic assumptions of standard statistics. In particular, whether a single link exists in an network or not is not independent from the existence of the other links (see section 2.3.1 Networks on personal level), in networks are row and column dependencies. Why does this matter? Because of this second number that always comes with the correlation coefficient—the significance value (level). In statistics a result is significant when there is a low probability that the result occurred by chance. And this significance test (statistical hypothesis tests)—the result is often referred to as *p*-value—requires independent lists of values as essential pre-condition. In summary, it is o.k. to calculate correlations between to networks matrices, but we won't have a clue whether the result is significant or not. Consequently, we need a better significance test that result in statistical guaranteed *p*-values. And this is exactly what people do when they run a QAP analysis.

6.3.4 QAP/MRQAP

David Krackhardt (1987b; 1988) presented the Quadratic Assignment Procedure (QAP) "for testing hypotheses in both simple and multiple regression models based on dyadic data, such as found in network analysis." (Krackhardt, 1988, p. 359) Krackhardt used the ideas from Hubert and Schultz (1976) to overcome the statistical dependencies problem of network data (see previous paragraph). But before we discuss this question in details, let us enumerate some statistical definitions. In stats, the *null hypothesis* is normally the assumption that two variables are uncorrelated. In the context of networks, these two variables represent two network matrices. In econometrics, temporally interdependent variables are called *autocorrelated*. For instance, the stock exchange price of a certain company at a certain day is a function of a lot of variables, one of which is the price at the day before our day of interest. In the context of networks, the interdependencies of data is called *structurally autocorrelated*.

When calculating correlations with network data the question is, does the network structure independently from the nodes cause the similarity between two networks or is the identity of the nodes relevant? To tackle this question, Krackhardt (1987b) suggested simulation approach called QAP. QAP is a graph-level test against the null hypothesis of uncorrelated data. The procedure itself is quite simple. First, you calculate the correlation of two networks. Krackhardt (1987b) used the network in Figure 6.4(a) to demonstrate the approach; the dependent network is without the dashed line and the

DRAFT



(a) Krackhardt's network: $r = 0.816$, $p = 0.027$

(b) Freeman's star: $r = 0.816$, $p = 0.204$

Figure 6.4: Correlation and QAP significance level for two networks

independent network is the same network but with the dashed line. Calculating Pearson's correlation for these two networks result in $r = 0.816$. Second, to get the p-value, the QAP approach comes into play. For the independent network (with the dashed line) we take the network matrix and we randomly permute the rows of the network matrix. Then we also change the order of the columns. The permutation of the columns is done identically to the rows. This procedure does not change the structure of the network but the identity of the nodes. We can get the same permuted network by simply changing the labels of the nodes. This random sampling of node label assignments or rows and column orders is called Monte Carlo simulation. Krackhardt did this for all possible 120 different permuted network and identified four networks that result in the same strong correlation (Krackhardt, 1988, p. 378). Therefore, the chance that the correlation is a random result triggered by the network structure is 3.3%—0.033 is the p-value of significance.

For very small networks it is possible to calculate all possible permutations. For five nodes there are 120 different permutations (= 5!). If the network is larger this is impossible. Even for a network with ten nodes there are already 3.6M different possible permutations. For practical applications, not all possible permutations are calculate but a rather small random sub-sample, e.g., 100 permutations. Consequently the p-value is then the percentage of permuted networks with a higher or equal correlation with the dependent network than the original independent network.

This is what we did in for both networks Figure 6.4. [2] The first network is

DRAFT

[2] For these examples we did the opposite of selecting a small random sub-sample of per- muted networks but selected 1,200 networks from the pool of 120 possible permutations.

DRAFT

the network that we have discussed in the previous paragraphs. The right network is the star structured network that Freeman (1979) used to describe the centrality measures. Both networks have five nodes and to both networks just one link is added to create the second network for comparison. The correlation coefficient is identical for both networks but the p-values are very different. The reason for the very bad (high) p-value for the second correlation is the structural dominance over the individual nodes—the structural autocorrelation. More than 20% of random permutations result in networks with equal or better correlation coefficient that the original independent network. Even $r = 0.816$, the result is not significant.

Multiple regression quadratic assignment procedure (MRQAP) is the approach for running regression analysis with more than two networks. Like regression analysis for regular statistical data, MRQAP describes the dependent network with a set of independent networks. The approach is very similar to QAP. We do not discuss details of MRQAP in this book; the interested reader will find more information about this topic at (Krackhardt, 1988) and (Dekker et al., 2007).

6.3.5 Exponential Random Graph (p^*) Models

Exponential Random Graph Models, also known as (p^*) models, are a family of statistical models that help analyze the structure and properties of social and other networks. There are other well known techniques to describe the structural properties of a network such as centrality, density etc. As opposed to these techniques which describe only the network for which they are measured, ERGMs try to describe all possible networks with the same statistical properties as the current one. The other alternative networks may or may not have the same structure, but the underlying statistical model that generates them is the same as the observed network. The networks that are being studied today are substantially larger in structure for example, the World Wide Web, Internet, communication networks, food web networks, etc which have millions and even billions of nodes. It is very hard to visualize the shape and structure of such networks, even with modern advances in computing technology. In absence of reliable visualization techniques, statistical modeling techniques can provide an answer by quantifying large networks. Networks with similar properties will have similar statistical models. This change in scale also makes traditional questions like, which nodes removal will affect the connectivity in the network the most, largely irrelevant. For large scale net-

This approach makes the resulting p-values very stable when re-running the analysis.

works a question which makes more sense is what percentage of nodes when removed would significantly affect network connectivity? Newman (2003). Statistical modeling of networks will allow us to answer this question.

Exponential random graph models have the following form:

$$P_\theta(X = x) = \frac{exp(\theta^t s(x))}{c(\theta)}$$

Where X is a random graph consisting on n nodes and x is the observed graph. The assumption of this model is that the structure of the observed graph x can be derived from a known vector of graph statistics, $s(x)$ and the associated vector of model parameters θ. The parameter $c(\theta)$, is a normalizing constant where

$$c(\theta) = \underset{\text{all possible graphs x}}{} \qquad exp(\theta^t s(x))$$

DRAFT

All exponential random graph models are of the form of the first equation, which describes a general probability distribution of graphs with n nodes. The particular probability for observing the graph x is dependent on its statistics $s(x)$ and on θ^t for all configurations in the model. Configurations include ties, triads and are these dependence assumptions are important because they pick different configurations as relevant to the model and also because they constrain the possible configurations possible in the model **?**. The simplest dependence assumption is the Bernoulli random graph distributions where the edges are assumed to be independent whereas more complex ERGMs incorporate node level effects (actor attributes) in the configurations. This flexibility in dependence assumptions allows for greater variability in the statistical modeling depending on the requirements and needs of the end user using the ERGM technique.

6.4 Detecting change

In the previous sections we learned how to compare specific networks with each other. For the last two sections of this chapter our focus shifts to analyzing many keyframes of a network over time. The initial example of this chapter in which we had thousands of emails for a time period of 81 days fits this criteria. Other examples of data are, the interaction of users in Social Media or co-publishing of scientific articles over decades, etc. In particular when you think on data from Social Media, you can create keyframes aggregating data on a daily level, or even for every hour. This results in a large number of keyframes, i.e. networks. Having all these networks, we are interested in quickly determine *that* (change detection) and *when* (change point identification) a change occurred in the network.

One approach to identify whether something interesting happened in these networks at some point in time make use of *Statistical Process Control*(McCulloh and Carley, 2011). Statistical Process Control (SPC) approaches are used in industrial production processes. Imagine a big machine producing lids for tubes of toothpaste. At the end of the production circle, right before the lids got packaged into boxes to be shipped to the toothpaste company, there are electrical sensors that take different measurements from every lid, e.g., weight, size, color. Each of this measure has an expected value that is pre-defined by the product engineers. As the instrument that collect these measures are very precise, it is almost impossible that the exact expected values are met. Therefore, a tolerance area of satisfactory is defined, i.e., a deviation from the expected value. We now use this idea of SPC for DNA.

DRAFT

6.4.1 Shewhart's Chart

The first SPC approach to detect change is Shewhart's chart (Shewhart, 1927). Recall the lid example from the previous paragraph. What do we need to analyze whether the lids have the correct size and color? We need to know what we are measuring (*statistic*) and when the measure is outside the expected area (*signal*). The first question is easy to answer. Various measures on node level or on network level that have been described in this and other books can be used to create statistics of interest, e.g., centrality measures. To create the statistic over time, we simply calculate the measure(s) of a network for each keyframe. The underlying assumption of change detection is that a change in an appropriate network measure is the result of a change in the underlying real-world network.

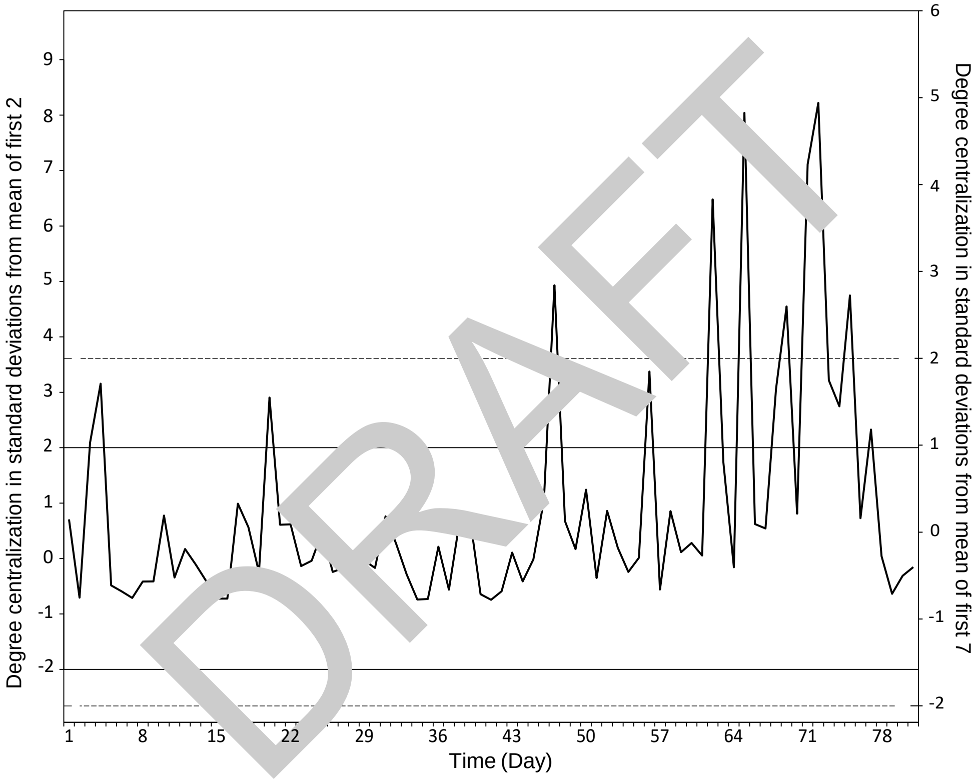
To illustrate Shewhart's approach for network data, we calculate total degree centralization Freeman (1979) over time for the e-mail network. The Curve in Figure 6.5 shows the result for every keyframe—please ignore the y-axes and the horizontal lines in the chart for now. For interpretation of this chart we need to know that day 1 of our data is a Thursday. With this knowledge we can identify a peak in centralization on the first weekend as well as on the third weekend and the consecutive days. In the second part of the data, the peaks get higher and the last 20 days of observation show a more dramatic up-and-down of degree centralization indicating short periods in which some people send and receive many more mails than the majority of the people in the network.

The second question concerning the *signal* is not trivially to answer. Looking at the curve in Figure 6.5 one could say that every peak is interesting. From a statistical point of view "a peak" is too fuzzy, we have to find a better definition. In particular, when we assume network data with hundreds or even thousands of keyframes. This brings us back to the question about what is the *expected* value in the context of networks? What is the expected degree centralization of our network, or what is the expected value of any network measure? Normally, we will not be able to answer these questions theoretically. Instead, we use parts of our empirical data to describe an *expected behavior* and calculate the deviation from these expectations. The tricky part is now to come up with a decision about which data to include. The left y-axis in Figure 6.5 uses just the first two days to calculate the boundaries that signal *unexpected* values. Therefore, 0.0 of the left y-axis is in the middle of the first two data points. The +/- values on this y-axis describe how many standard deviations the data points are away from this average value. If we assume an expected area of +/- 2.0 standard deviations (continuous lines), the first two peaks (and also the later peaks) are identified as spikes of our statistic.

Maybe we already know that, because of some characteristics of this system, total degree centralization peaks on weekends and we want to incorporate this into the expected behavior of the change detection (we talk more about these periodicities in the next section). In this case, our observation period could be seven days. The right y-axis and the dashed lines show average and standard deviations calculated on the first seven data points. You can see that now the first peaks are within the boarders of expected behavior. The third larger peak is the first that triggers the signal of unexpected behavior. The last 20 days are still covered by this version of detecting change.

This straightforward example already illustrates the main issue of change detection for network measures—the definition of the expected behavior of the system which is the primary trigger for signaling deviating behavior of the system. First of all, this a decision of trade-off between false positive and rapid detection, i.e., a narrow area of expected behavior is able to identify small change but results in a lot of *false alarms* while a broader tolerance region produces less false alarms but includes a higher risk of missing interesting changes in the network. In the context of SPC, Page (1961) suggests a second line on a lower level—the *warning line*.

DRAFT



DRAFT

Figure 6.5: Shewhart chart for degree centralization of e-mail data

6.4.2 Cumulative Sum (CUSUM)

Shewhart's control chart Shewhart (1927), that was introduced in the last section, focuses on the identification of single outliers in the data. The cumulative sum (CUSUM) approach is good for detecting smaller but constant change. A simple version of CUSUM is the following. Imagine a control chart like in Figure 6.5. Instead of one *action line* that signals a deviation of the system, we could draw two or more lines, some of which even closer to the expected average value. Passing the first control line results in 1 point, passing the second line results in 2 points, and so forth. A data value within the expected area results in, e.g., -2 points. If we now setup the system in a way that reaching 3 (or more) points triggers the deviation signal than this threshold can be reached with one or two big deviations or a larger number of smaller deviations.

A more generalized version of CUSUM was introduced by (Page, 1961). The CUSUM chart consists of two lines. Let us first focus on the first line that controls for positive deviations, i.e., the observed data points x_i are bigger than the expected value k. For instance, the first data points of our observed measures are above k. CUSUM accumulates now the deviation from k by calculating $S_2 = (x_1 - k) + (x_2 - k)$. It is clear that this line is going upwards as long as the data points are larger than the expected value. In case the data points are smaller than k, the CUSUM value gets smaller but the accumulated deviation is still positive. If this line hits now, e.g., as a result of many small positive deviations, a pre-defined threshold, the change signal is triggered. In case the threshold is not reached but instead the CUSUM gets smaller and smaller and finally undershoots zero, the positive lines is set to zero and the negative CUSUM line starts to accumulate negative deviations.

In Figure 6.6 we used the same email data that was used previously in this chapter. As we have seen before that the weekends are very different in our data, we deleted the Saturday and Sunday networks from our Dynamic Meta Network. For the CUSUM analysis we looked at the number of *Agents* in the network. The number of nodes that are involved in email conversation (send and/or receive) is visualized with the gray line in Figure 6.6. The y-axes marks the amount of standard deviations from the mean based on the first five observations. If you look at the gray curve in of the statistic you may think that values go up and down around the zero line marking the average. Looking at the cumulative sums reveals a very different picture. Even the one day peak at day 20 is followed by a rather stable period of 20 days, the values of this period are almost completely above the average—the constantly raising increase lines indicates this artifact of our statistic. Moreover, when

DRAFT

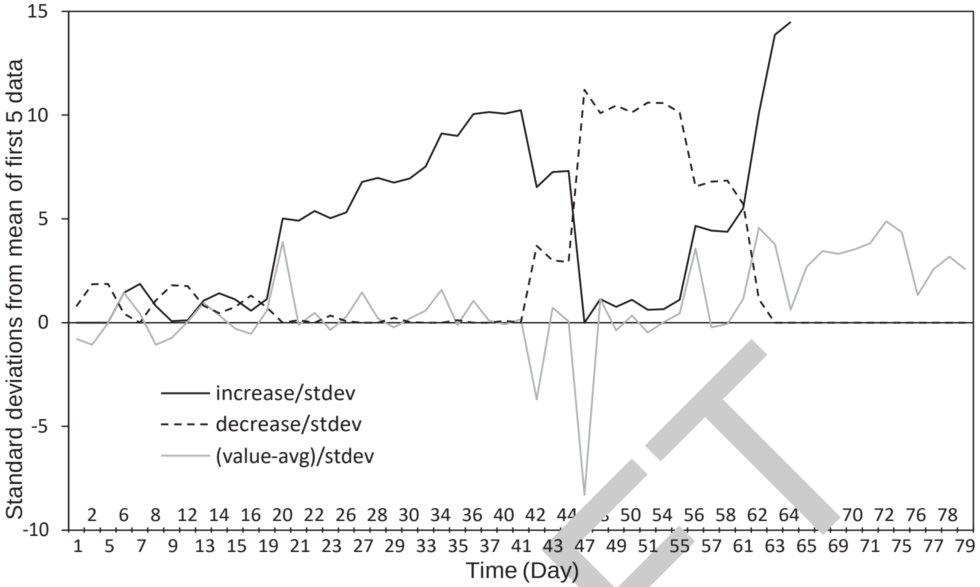


Figure 6.6: Cumulative sum chart

DRAFT

we look at the point at which the increase line leaves the zero line we can identify the change point of the statistic (day 12). CUSUM The very high values of the last 20 days of the data burst the bonds of the chart. Therefore, we stop drawing the increase CUSUM at this point.

Shewhart's control chart and Page's CUSUM are straightforward approaches that are easy to understand and to calculate. The decision about which approach to use is based on the question of whether you are more interested in detecting all spikes of deviation or you are looking for constant (and probably smaller) changes. Another advantage of CUSUM is the built-in change point detection. If you are interested in statistically more elaborated approaches to Statistical Process Control we want to point your attention to McCulloh (2009) for an introduction to this topic in the context of network analysis.

6.5 Periodicities

Many real world network data over time, in particular data based on communication like the e-mail data that we have used in the previous sections, show periodic patters, i.e., a very similar up-and-down over time. For instance, it makes perfects sense that e-mail data has a weekly periodicity as people work less on the weekends. In addition a monthly periodicity could indicate reporting a reporting structure that create more communication at the end of the month independently from the weekday. This changing behavior creates similar networks on a periodic intervals that probably result in similar network statistics on network or node level. In real-world data the weekly and the monthly periodicity overlay each other. Actually, additional periodicities based on project cycles can make it even harder to identify the underlying periodicities. Nevertheless, identifying the periodic behavior of ones data can be a very interesting step of analysis.

A very powerful but all but trivial approach to describe periodicities automatically is *Fourier analysis*. Fourier analysis decomposes a signal into a sum of sine waves. What does this mean? A sine wave is function $a \cdot sin(t/b)$.

Figure xx with 3 or four difference sine waves: $2 \cdot sin(t/\pi)$

Figure xx shows how different sine waves can be used to describe more complex periodicities. The third sine wave is the product of the first two sine waves. Adding more sine waves can create any possible curve—theoretically we could even create the rectangular signal *nunu* with an infinite number of sinewaves.

figure: sum of 2 sine waves (from AHFE workshop)—write sin(xx) to the figure.

Spectral analysis.

Use Sine waves to describe periodicity

Detect temporal regularities and cycles

Power spectral density (PSD): How much power/energy lays in which periodicity?

frequency domain Dominant Frequencies: Shows just frequencies that are greater than two standard deviations from the mean frequency

Period Plot: applying an inverse Fourier transform to the dominant frequencies

6.6 Problem set

1. What are the two ways of describing change in network and which one is used normally in network analysis?

2. What two-mode network can be used to show the trail of Julius Caesar in the tragedy?

DRAFT

3. What data is used when network distances are calculated?

4. If you assume two networks that have 80% of the nodes overlapping. Does the *union* or the *intersect* of the networks have more nodes? Why?

5. When we want to compare the centrality of an actor over time. What data do we use to create the line diagram of importance over time? How do we get this data?

6. What is *false positive* and *false negative*?

7. What is the advantage of CUSUM compared to Shewhart's chart?

8. What is the basic idea of fast Fourier transformation?

9. How is it possible that we can calculate Pearson correlation for two network matrices?

10. Look at the overtime calculations in chapter 2 that show the importance of *Agents* in the tragedy of Julius Caesar. What network level metrics could we use to analyze change in this network?

11. *If your want to calculate Hamming distance from the network of an organization over time with fluctuating members; what is the problem? How can you overcome this problem?

12. *Aggregate (union) all five acts of the Julius Caesar *Agent* x *Agent* network. Calculate degree, closeness, and betweenness centrality for the aggregated as well as for the per-act networks. Discuss the differences of the calculations. What are the advantages and disadvantages of both methods?

13. *What is the problem with network correlation and how can we overcome this problem?

14. *We used *Freeman's star* network in this chapter to show a bad example for the QAP procedure. Find another network consisting of five nodes for which adding a single link results in bad QAP significance level.

15. *Find a scenario that satisfies these conditions: The node level metric is completely different and the network level metric is identical.

16. *Visualize the overlapping trails of three main *Agents* of the Julius Caesar network.

DRAFT

17. **Enumerate the four levels of comparison and describe how change on each of these levels can be calculated. Find a network with at least two keyframes and calculate a change metrics for every level. Discuss the your results in the context of your network data.

18. **Figure out why correlation of two binary network matrices is problematic. You can discuss this question with a statistician.

19. **Find and example for exponential random graph models in the literature and understand what was accomplished in this work.

20. **Find a network online or create a network that has at least ten keyframes. Calculate degree centralization for every keyframe and calculate the Shewhart's chart. Experiment with different numbers of networks that are in control. Can you find useful results?

DRAFT

DRAFT

Chapter 7

Network Evolution and Diffusion

7.1 Diffusion of apples, ideas and beliefs

7.1.1 Random networks and stylized networks

7.1.2 Diffusion of innovation

7.1.3 Epidemic concepts

7.2 Agent-based dynamic-network computer simulations

7.2.1 Models

7.2.2 Models for diffusion processes

7.3 Evolution of networks

DRAFT

DRAFT

Chapter 8

Extracting Networks from Texts

DRAFT

It has become fast, cheap, and easy to collect and store large amounts of natural language text data. New text data is created every second on the internet as hundreds of million of user interact on social media platforms with each other. But also traditional media is more and more available since archival text data such as newspapers and books are being converted to digital forms daily. The increasing availability of these data has exacerbated the need for techniques, measures and tools for automated knowledge discovery and reasoning about text. The family of methods developed to address this issue is collectively known as *Relational Text Analysis* (RTA). Or, in the context of networks it is often referred to as *Network Text Analysis* (Carley, 1997; Danowski, 1993; van Cuilenburg, Kleinnijenhuis and Ridder, 1986). While text data does not automatically entail a network analysis, text is a ready source of new information about relationships between entities and additional attributes for single entities.

The goal of this chapter is to introduce you to the applicability and usage of text analysis in the domain of network analysis. You will learn how to extract relevant information and structured data from text data in an efficient and systematic fashion, how to perform appropriate analysis on the extracted data, and how to interpret and evaluate your results. The secondary goals are to understand how the different choices that you make throughout this process impact your results. We will describe the basics for how to distill relevant information as well as one-mode and multi-mode network data from texts. We will also discuss what metrics you would use in analyzing the extracted information and network data.

8.1 Analyzing texts

8.1.1 Content analysis

8.1.2 tf*idf

8.2 Text processing

8.2.1 Deletion

8.2.2 Thesauri

8.2.3 Concept lists

8.2.4 Bi-grams

8.2.5 Stemming

8.3 From texts to networks

8.3.1 Keyword in context

8.3.2 Windowing

8.3.3 Extracting meta-networks from texts

DRAFT

Chapter 9

The Future of Dynamic Network Analysis

DRAFT

DRAFT

Appendix A

SNA Measures Glossary

DRAFT

Dynamic network analysis uses a lot of different measures to gain better insights into network structure and dynamic. In the chapters of this book we discussed various measures to identify important nodes, find group structures, detect change, etc. We have not defined these measures mathematically or algorithmically so far. This is what you can find in this chapter. At the following pages you will find a lot of measures including a short description, the reference of the paper where the measure was originally presented, as well as the equation or algorithm describing the actual calculation of the measures.

Scaling a measures means to transform the results of a certain calculation into the range of 0 to 1, with 0 being the smallest possible value and 1 indicating nodes having the maximum possible value. This is important to make results of different networks comparable. The basic idea of scaling network measures is to find the maximum possible value and divide by this number, e.g. in a network with N nodes a single node can have $N-1$ maximum possible neighbors. Degree centrality, therefore, counts the number of nodes, a single not is connected to and divides this number by $N-1$. In the following equations you will find an apostrophe to indicate scaled measures, e.g. C^t stands for scaled degree centrality. Consequently, C is unscaled degree cen$_D$trality, which is the number of neighbors of a node. For some measures we offer the unscaled version and the maximum possible value. In this case, to get the scaled measure you have to divide the unscaled with the maximum value:

$$C^t = \frac{C}{C^{max}}$$

A.1 Notations

A.1.1 Node Classes

A single network matrix is notated with capitalized letters. We use different letters to refer to different node classes. These are the abbreviations for the node classes:

A Agent	*T* Task	*R* Resource	*K* Knowledge	*E* Event
C Action	*O* Organization	*L* Location	*X* Role	*B* Belief

Table A.1: Node Classes

A.1.2 Matrices

For some of the measures we need matrix notation to describe the measure. Table A.1.2 gives an overview of these notations as well as short descriptions.

A.2 Standard network measures

A.2.1 Degree Centrality

Citation: Freeman (1979)
Description: Degree centrality measures the number of other nodes that one node is connected to. Depending on the network, high degree centrality indicates a highly active agent or an agent known by a lot of other agents, etc.

$$C_D(i) = \sum_{j>i} w_{j,i} \qquad with \quad C_D^{max} = N - 1$$

A.2.2 Closeness Centrality

Citation: Sabidussi (1966), Freeman (1979)
Description: Closeness centrality measures the nearness (as opposite from the distance) from an agent to all other agents. Agents having a high closeness

DRAFT

Notation	Description
A	Capitalized letters represent one mode (squared) network matrices, e.g. the connections between *Agents*
A	Capitalized bold letters represent a matrix which is the result of a calculation
AR	Two mode network matrix, e.g. the connections between *Agent* and *Ressources*
$\|A\|$	Dimension of a squared matrix, i.e. the number of nodes in the network
$A(i, j)$	The entry in the i^{th} row and j^{th} column of the matrix
$A(i, :)$	The i^{th} row vector of a matrix
$A(:, i)$	The j^{th} column vector of a matrix
(A)	The sum of all elements of a matrix
A^t	The transpose of a matrix, i.e. rows and columns are swapped
$\sim A$	For binary matrix, $A(i,j) = 1$ if $A(i,j) = 0$, i.e. swaps 0 and 1,
$A \circ A$	Element-wise multiplication of two matrices, i.e. $C = A \circ B \Rightarrow C(i,j) = A(i,j) \cdot B(i,j)$
card(Set)	The cardinality of a set, i.e. $\|Set\|$

Table A.2: Matrix notations

score have short distances to all other nodes. This is important for the availability of knowledge and resources. $d(i, u)$ is the path distance from node i to node u.

$$C_C(i) = \frac{1}{_{i/=u}\ d(i, u)} \qquad with \quad C_C^{max} = \frac{1}{N-1}$$

A.2.3 Betweenness Centrality

Citation: Anthonisse (1971), Freeman (1977, 1979)
Description: Betweenness centrality measures the amount an actor is in an intermediate position between other nodes. High between actors connect different groups and have control over the flow of information in a network. $g_{u,v}$ is the number of shortest paths between two nodes u and v while $g_{u,v}(i)$ is the number of shortest paths including node i.

DRAFT

$$C_B(i) = \underset{u<v}{}\frac{g_{u,v}(i)}{g_{u,v}} \quad with \quad C_B^{max} = \frac{N^2-3N+2}{2}$$

A.2.4 Eigenvector Centrality

Citation: Bonacich (1972)
Description: Eigenvector centrality is based on eigenvector calculation in linear algebra. Agents have a high eigenvector score if they are important and connected to other important agents. Let W be the network matrix, λ be the largest eigenvalue of the adjacency matrix W, and C_E the corresponding eigenvectors. The Eigenvector centrality of a node i, $C_E(u)$ is defined as the linear combination of the eigenvector centrality of its neighbors:

$$C_E(i) = \frac{1}{\lambda}\underset{u}{}\ w_{i,u}C_E(I) \quad with \quad C^{max} = \sqrt{0.5}$$

where λ is a constant. We can rewrite the equation as:

$$\lambda C_E = W \cdot C_E$$

DRAFT

A.2.5 Clustering Coefficient

Citation: Watts and Strogatz (1998)
Description: The Clustering coefficient measures the local density of every agent. Agents with a high clustering coefficient are connected to neighbors which are more likely connected to each other. For a vertex i with k_i neighbors, these neighbors can have at most $k_i(k_i-1)/2$ edges. The clustering coefficient for the node i is the number of actual links between the k_v neighbors divided by the maximum possible number:

$$CC(i) = \frac{2\cdot|W_{u,v}|}{|N_i|(N_i-1)} \quad with \quad e_{i,u}, e_{i,v} \in E$$

A.3 Grouping algorithms

A.4 Change measures

A.5 Network text algorithms

DRAFT

DRAFT

Appendix B

Two-Mode Network Measures

DRAFT

B.1 Quantity

Quantity measures count or average the entries of a matrix.

B.1.1 Degree

Counting the row or column entries of a two mode network. In case of an AK network the normalized row degree **d** for an agent i is defined as follows:

$$\mathbf{d}_i = \frac{{}_{j=1}^{|K|} AK(i,j)}{|K|}$$

Level: Node
Reference: Freeman (1979), Wasserman and Faust (1994), Borgatti and Everett (1997)
Current ORA measures: columnDegreeCentrality, inDegreeCentrality, outDegreeCentrality, rowDegreeCentrality, columnCount, rowCount, edgeCount, capability

B.1.2 Load

On network level, a single value for the entire network is calculated through averaging link values. Load is a network level concept that identifies the average amount of, e.g. Knowledge, per agent. Knowledge load **l** is defined as

$$\mathbf{l} = \frac{AK}{|A|}$$

Level: Network
Reference: Carley (2002)
Current ORA measures: knowledgeLoad, resourceLoad, density, rowBreadth, columnBreadth.

B.2 Variance

Variance measures create network level indices that describe the distribution of connections in networks.

B.2.1 Centralization

On way to describe the variance in a network is through calculating network centralizations based on the results of centrality measures. If $c(p_i)$ is the centrality score of node i and $c(p*)$ is the largest value for any node in the network, then the network centralization **c** is defined as

$$\mathbf{c} = \frac{\sum_{i=1}^{n}[c(p*)-c(p_i)]}{\max \sum_{i=1}^{n}[c(p*)-c(p_i)]}$$

Level: Network
Reference: Freeman (1979)
Current ORA measures: columnDegreeCentralization, inDegreeCentralization, outDegreeCentralization, rowDegreeCentralization

B.2.2 Diversity

Diversity is a group measure and results in a single value for the whole network. It measures whether the knowledge of the people in an organization is rather equally distributed or concentrated. This is the Herfindahl-Hirshman index applied to column sums of, e.g. the AK network. For every knowledge we calculate

$$w_k = \sum_{i=1}^{|A|} AK(i,k)$$

with

$$W = \sum_{k=1}^{|K|} w_k$$

DRAFT

then

$$\mathbf{d} = 1 - \sum_{k=1}^{|K|} \left(\frac{w_k}{W}\right)^2$$

Level: Network
Reference: Hirschman (1945)
Current ORA measures: knowledgeDiversity, resourceDiversity

B.3 Correlation

Correlation measures create a matrix **A** that describe similarities/dissimilarities between all pair of agents. For some measures, this dyadic matrix is the result. For other measures, node level or network level indices are calculated.

B.3.1 Similarity

The degree of similarity between two agents based on shared knowledge. Each agent computes to what degree the other agents know what they know. Let

$$\mathbf{M} = AK \cdot AK^t$$

and

$$w(i) = \mathbf{M}(i,:) \text{ for } 1 \le i \le |A|$$

then the relative similarity **S** between agents ***i*** and ***j*** is

$$\mathbf{S} = \mathbf{M}(i,j)/w(i)$$

The relative similarity **s** for an agent ***i*** is the average of the non-diagonal elements of row ***i*** of **S**

$$\mathbf{s}_i = \frac{\sum_{j=1, j \neq i}^{|A|} \mathbf{S}(i,j)}{|A|-1}$$

Level: Node
Reference: Carley (2002)
Current ORA measures: relativeCognitiveSimilarity, cognitiveSimilarity, relativeSimilarity, correlationSimilarity

DRAFT

B.3.2 Distinctiveness

Measures the degree to which each pair of agents (i, j) has complementary knowledge, expressed as the percentage of total knowledge. In effect, this is the exclusive-OR of the knowledge vectors. Cognitive distinctiveness **d** for a pair of agents (i, j) as well as (j, i) where $\boldsymbol{i} /= \boldsymbol{j}$ is

$$\mathbf{d}_{i,j} = \frac{\sum_{k=1}^{|K|} (AK_{i,k} \cdot \sim AK_{j,k}) + (\sim AK_{i,k} \cdot AK_{j,k})}{|K|}$$

Level: Dyad
Reference: Carley (2002)
Current ORA measures: relativeCognitiveDistinctiveness, cognitiveDistinctiveness, correlationDistinctiveness

B.3.3 Resemblance

Measures the degree to which each pair of agents has the exact same knowledge. Cognitive resemblance **r** for a pair of agents (i, j) as well as (j, i) where $\boldsymbol{i} /= \boldsymbol{j}$ is

$$\mathbf{r}_{i,j} = \frac{\sum_{k=1}^{|K|} (AK_{i,k} \cdot AK_{j,k}) + (\sim AK_{i,k} \cdot \sim AK_{j,k})}{|K|}$$

Level: Dyad
Reference: Carley (2002)
Current ORA measures: relativeCognitiveResemblance, cognitiveResemblance, correlationResemblance

DRAFT

B.3.4 Expertise

The degree of dissimilarity between agents based on shared knowledge. Each agent computes to what degree the other agents know what they do not know. The relative expertise matrix **E** is defined as follows:

$$\mathbf{E}(\sim AK \cdot AK^t) \qquad \textit{with } \mathbf{E}(i, i) = 0$$

normalize **E** by its row sum

$$\mathbf{E}(i, :) = \frac{\mathbf{E}(i, :)}{\mathbf{E}(i, :)}$$

The relative expertise **e** for agent i is

$$\mathbf{e}_i = \frac{\sum_{j=1,j\neq i}^{|A|} \mathbf{E}(i,j)}{|A|-1}$$

Level: Node
Reference: Carley (2002)
Current ORA measures: relativeCognitiveExpertise, cognitiveExpertise, relativeExpertise, correlationExpertise

B.4 Specialization

These measures identify agents that have either exclusive or redundant connections to other node class entities.

B.4.1 Exclusivity

Detects agents who are exclusively connected to elements of other node classes. The knowledge exclusivity index **x** for an agent i is defined as follows:

$$\mathbf{x}_i = \sum_{j=1}^{|K|} \left[AK(i,j) \cdot \exp\left(1 - \sum AK(:,j)\right)\right]$$

Level: Node
Reference: Ashworth and Carley (2006)
Current ORA measures: knowledgeExclusivity, resourceExclusivity, taskExclusivity, exclusivityComplete, exclusivity

B.4.2 Redundancy

Knowledge is redundant if there are different agents sharing the same knowledge. Redundancy is a network level measure and returns the average number of redundant agents per knowledge. For every column j in the AK matrix we calculate

$$d_j = \max\left[0, \sum AK(:,j) - 1\right]$$

Knowledge redundancy **r** is consequently

$$\mathbf{r} = \frac{\sum_{j=1}^{n} d_j}{|K|}$$

DRAFT

with

$$\mathbf{r}_{\max} = (|A| - 1)$$

Level: Network
Reference: Carley (2002)
Current ORA measures: columnRedundancy, rowRedundancy, knowledgeRedundancy, accessRedundancy, resourceRedundancy, assignmentRedundancy

B.4.3 Access

An access index identifies connections to critical knowledge, resources, etc. The knowledge access index **a** first identifies actors that have exclusive connections to knowledge. If such an agents is in addition connected to just one other actor in the social network then both agents have critical access. For every agent a set of exclusive knowledge is calculated in case the agent is just connected to one other agent:

$$K_i = \{k | AK(i,k) \wedge (\quad AK(:,k) = 1) \wedge (\quad A(i,:) = 1)\}$$

For agent i **a** is binary and defined as

$$\mathbf{a}_i = ((K_i /= \emptyset) \vee (\exists j | K_j /= \emptyset \wedge A(j,i) = 1))$$

Level: Node
Reference: Ashworth and Carley (2006)
Current ORA measures: knowledgeAccessIndex, resourceAccessIndex

DRAFT

Appendix C

Multi-Mode Network Measures

C.1 Quantity

C.1.1 Degree

Total number of people reporting to an agent, plus its total knowledge, resources, and tasks can be considered as *Personnel Costs*:

$$\mathbf{c}_i = \frac{\sum_{j=1, j\neq i}^{|A|} AA(j,i) + \sum AK(i,:) + \sum AR(i,:) + \sum AT(i,:)}{(|A|-1)+|K|+|R|+|T|}$$

A very similar measure is *Socio Economic Power* that measures the power of *Agents* based on access to knowledge, resources, and tasks. It is defined like *Personnel Costs* but without the *Agents* parts of the equation.

Level: Node
Reference: Ashworth and Carley (2003), Carley (2004)
Current ORA measures: personnelCost, agentSocioEconomicPower

C.1.2 Load

Complexity is a meta-network measures of load. It calculates the density of the meta-matrix that results from concatenating all available networks:

DRAFT

$$\mathbf{c} = \frac{card\{\forall i, \boldsymbol{j} \in A | i /= \boldsymbol{j} \wedge AA(i,j) > 0\} + card\{\forall i \in A \wedge \forall t \in T | AT(i,t) > 0\} + \cdots}{|A|(|A|-1) + |AT| + \cdots}$$

Level: Network
Reference: Wasserman and Faust (1994)
Current ORA measures: complexity

C.2 Coherence

Coherence measures analyze to what extent requirements are accomplished by the actual allocation of agents to tasks. The three different concepts of this group describe coherence from different perspectives. *Congruence* measures the level of conformance, *need* algorithms focus on missing *Knowledge* or *Resources*, and *waste* algorithms identify surplus *Knowledge* or *Resources* compared to the actual needed ones for completing tasks. *Performance* looks at *Tasks* which can be completed.

C.2.1 Congruence

Measures the similarity between what *Knowledge* is assigned to tasks via agents, and what *Knowledge* is required to do *Tasks*. Perfect congruence occurs when *Agents* have *Knowledge* when and only when it is needful to complete their *Tasks*. Let **KT** be the matrix representing the *Knowledge* assigned to *Tasks* via *Agents*

$$\mathbf{KT} = AK^t \cdot AT$$

then then *Knowledge Congruence* is the proportion of *correctly* assigned *Knowledge*

$$\mathbf{c} = \frac{\text{card}\{(i,j)|[(\mathbf{KT}(i,j) > 0) = (KT(i,j) > 0)]\}}{|K| \cdot |T|}$$

Another more elaborated measures is *Communication Congruence* that measures to what extent *Agents* communicate when and only when it is needful to complete *Tasks*. We assume that *Agents* ***i*** and ***j*** must reciprocally communicate if at least one of the following is true. a) if ***i*** is assigned to a *Task s* and ***j*** is assigned to a *Task t* and *s* directly precedes *Task t* (handoff). b) if

DRAFT

i is assigned to a *Task s* and *j* is also assigned to *s* (co-assignment). c) if *i* is assigned to a *Task s* and *j* is not, and there is a *Resource r* to which *Agents* assigned to *s* have no access but *j* does (negotiation to get needed *Resource*). These three cases are computed as follows:

a) $\mathbf{H} = AT \cdot T \cdot AT^t$

b) $\mathbf{C} = AT \cdot AT^t$

c) $\mathbf{N} = AT \cdot Z \cdot AR^t \quad \textit{with } \mathbf{Z}(t,r) = \begin{cases} 1 & \text{if } [AT^t \cdot AR - RT^t](t,r) < 0 \\ 0 & \text{otherwise} \end{cases}$

Let **Q** be the reciprocal communication matrix that is required

$$\mathbf{Q}(t,r) = \begin{cases} 1 & \text{if } [(\mathbf{H}+\mathbf{C}+\mathbf{N}) + (\mathbf{H}+\mathbf{C}+\mathbf{N})^t](i,j) > 0 \\ 0 & \text{otherwise} \end{cases}$$

then *Communication Congruence* is, similar to *Knowledge Congruence*, the degree to which actual communication differs from that which is needed to do tasks

$$\mathbf{c} = \frac{\text{card}\{(i,j) | [(\mathbf{A}(i,j) > 0) = (\mathbf{Q}(i,j) > 0)]\}}{|A| \cdot (|A| - 1)}$$

Level: Network
Reference: Carley (2002)
Current ORA measures: communicationCongruence, knowledgeCongruence, resourceCongruence, socialTechnicalCongruence (2 mode)

C.2.2 Needs

Task Knowledge Needs compares the *Knowledge* requirements of each *Task* with the *Knowledge* available to the *Task* via *Agents* assigned to it. It is similar to *Knowledge Congruence*, but quantifies only the under supply of *Knowledge* to *Tasks*. Let **S** be the *Knowledge* supplied to *Tasks* via assigned *Agents*.

$$\mathbf{S} = AT^t \cdot AK$$

then *Task Knowledge Needs* for *t* is defined as

$$\mathbf{n}_t = \frac{{}^{|K|}_{k=1} \; KT^t(t,k) \cdot (\mathbf{S}(t,k) = 0)}{{}^{|K|}_{k=1} \; KT^t(t,k)}$$

DRAFT

Organization Needs measures are the corresponding network level measures, e.g. the *Knowledge* that *Agents* lack to do their assigned *Task* expressed as a percentage of the total *Knowledge* needed by all *Agents*:

$$\mathbf{n} = \frac{\sum_{t=1}^{|T|} \sum_{k=1}^{|K|} KT^{t}(t,k) \cdot (\mathbf{S}(t,k) = 0)}{KT}$$

Level: Node, Network
Reference:Lee and Carley (2004)

Current ORA measures: communicativeNeeds (communication), congruenceAgentKnowledgeNeeds, congruenceOrgAgentKnowledgeNeeds, congruenceOrgTaskKnowledgeNeeds, congruenceTaskKnowledgeNeeds, congruenceAgentResourceNeeds, congruenceOrgAgentResourceNeeds, congruenceOrgTaskResourceNeeds, congruenceTaskResourceNeeds, knowledgeUnderSupply, resourceUnderSupply

C.2.3 Waste

Waste measures focus on the oversupply part of congruence. *Task Knowledge Waste* counts the number of skills supplied to a *Task* via *Agents* that are not required by it expressed as a percentage of the total skills required for the *Task*. This measure results in a value for every *Task*. Let, again, **S** be the *Knowledge* supplied to *Tasks* via assigned *Agents*.

$$\mathbf{S} = AT^{t} \cdot AK$$

then *Task Knowledge Waste* for *t* is defined as

$$\mathbf{w}_{t} = \frac{\sum_{k=1}^{|K|} S(t,k) \cdot (\sim KT^{t}(t,k))}{\sum_{k=1}^{|K|} S(t,k)}$$

Organizational Waste measures are the corresponding network level measures, e.g. the *Knowledge* supplied to *Tasks* via *Agents* that are not required by them, expressed as a percentage of the total skills needed by all *Tasks*:

$$\mathbf{w} = \frac{\sum_{t=1}^{|T|} \sum_{k=1}^{|K|} S(t,k) \cdot (\sim KT^{t}(t,k))}{S}$$

Level: Node, Network
Reference: Lee and Carley (2004)

DRAFT

Current ORA measures: congruenceAgentKnowledgeWaste, congruenceOrgAgentKnowledgeWaste, congruenceOrgTaskKnowledgeWaste, congruenceTaskKnowledgeWaste, congruenceAgentResourceWaste, congruenceOrgAgentResourceWaste, congruenceOrgTaskResourceWaste, congruenceTaskResourceWaste

C.2.4 Performance

Knowledge Based Task Completion calculated the percentage of tasks that can be completed by the agents assigned to them, based solely on whether the agents have the requisite *Knowledge* to do the *Tasks*. First, we find the *Tasks* that cannot be completed because the *Agents* assigned to the *Tasks* lack necessary *Knowledge*:

$$\mathbf{N} = (AT^{t} \cdot AK) - KT^{t}$$

Then we calculate the set of *Tasks* which cannot be completed:

$$\mathbf{S} = \{t | t \in T \wedge \exists k : \mathbf{N}(t, k) < 0\}$$

Knowledge Based Task Completion is then the percentage of tasks that could be completed:

$$\mathbf{c} = \frac{|T| - |S|}{|T|}$$

Level: Network
Reference: Carley (2002)
Current ORA measures: resourceTaskCompletion, overallTaskCompletion, knowledgeTaskCompletion, performanceAsAccuracy

DRAFT

C.2.5 Workload

Workload measures look for the *Knowledge* or *Resource* an *Agent* uses to perform *Tasks* to which it is assigned. The *Potential Knowledge Workload* is defined as

$$\mathbf{w}_{i} = \frac{(AK \cdot KT)(i, :)}{KT}$$

while the *Actual Knowlede Workload* also takes the actual *AT* network into account:

$$\mathbf{w}_{i} = \frac{(AK \cdot \underset{T}{K} \cdot AT^{t})(i, i)}{KT}$$

Level: Node
Reference: Carley (2002)

Current ORA measures: knowledgeActualWorkload, resourceActualWorkload, actualWorkload, knowledgePotentialWorkload, resourcePotentialWorkload, potentialWorkload

C.2.6 Negotiation

The extent to which *Agents* need to negotiate with each other because they lack the *Knowledge* to complete their assigned *Tasks*. The *Negotiation* measure computes the percentage of tasks that lack at least one *Knowledge*. First, a *TK* congruence matrix is calculated with

$$\mathbf{C} = (AT^t \cdot AK) - KT^t$$

to get proportion of unassigned *Tasks*:

$$\mathbf{n} = \frac{\text{card}\{t|t \in T \wedge \exists k : \mathbf{C}(t,k) < 0\}}{|T|}$$

Level: Network
Reference: Carley (2002)
Current ORA measures: knowledgeNegotiation, resourceNegotiation

C.3 Substitution

This group is rather diverse and include concepts and measures describing to which extent *Agents* can be substituted base on identical *Roles* or *Agents* need to negotiate with each other as well as *Knowledge* and *Resources* that is used or reused to perform *Tasks*.

C.3.1 Availability

Availability measures use the node class *Roles*. *Role Based Knowledge Availability* computes the number of roles that an agent is qualified to have based on knowledge requirements. *Overall Role Based Availability* computes the degree to which agents are available to do tasks based on their access to knowledge and resources and roles that are needed to do the tasks. The later is defined as ...

$$\mathbf{a} \ =?AK??AR??AX?$$

DRAFT

Level: Network
Reference: Behrman
Current ORA measures: roleKnowledgeAvailability, roleResourceAvailability, knowledgeBasedRoleAvailability, organizationalAvailability, resourceBasedRoleAvailability, overallRoleAvailability

C.3.2 Reuse

Knowledge utilization can be calculated with *Reuse* measures of *Knowledge* or *Resources*. The interested question is whether e.g. *Knowledge* that is already available in an Organization can be reused to accomplish *Tasks*. Let **TAT** be a matrix connecting *Tasks* to which identical *Agents* are assigned to:

$$\mathbf{TAT} = AT^{t} \cdot AT$$

Then *Knowledge Based Omega* is the proportion of *Knowledge* used in *Tasks* that has been already used in previous *Tasks*:

$$\omega = \frac{((TT^{t} \circ TAT) \cdot KT^{t}) \circ KT^{t}}{KT}$$

Level: Network
Reference: Carley et al. (2000)
Current ORA measures: knowledgeOmega, resourceOmega

C.4 Control

Control measures are the most complex measures as they combine a couple of calculations into a single measure. Control measures describe to which extent an *Agent* is either important (central) for an entire meta-network or *Agents* have similar perspectives of the network.

C.4.1 Demand

Cognitive Demand measures total amount of cognitive effort expended by each agent to communicate, performs its tasks, etc. The *Cognitive Demand* for an agent *i* is an average of terms, each of which measures an aspect of its cognitive demand. Each term is normalized to be in [0,1]. The number of terms depends on the available input networks. The first three terms cover

DRAFT

the number of entities *Agent i* is connected to in different networks:

$$\mathbf{d}_i = \frac{^{|A|}_{j,j/=i} AA(i,j)}{|A|-1}$$

$$\mathbf{d}_i^2 = \frac{[AT](i,:)}{|T|}$$

$$\mathbf{d}_i^3 = \frac{[AR](i,:)}{|R|}$$

$$\mathbf{d}_i^4 = \frac{[AK](i,:)}{|K|}$$

The next terms cover the number of *Agents* assigned to the same *Tasks* as *i* as well as the *Resources* and *Knowledge* needed by *i* to complete her *Tasks*:

$$\mathbf{d}_i = \frac{\mathbf{ATA}(i,:) - \mathbf{ATA}(i,i)}{|T|(|A|-1)} \quad \text{with } \mathbf{ATA} = AT \cdot AT^t$$

$$\mathbf{d}_i^6 = \frac{\mathbf{ATR}(i,:)}{|T| \cdot |R|} \quad \text{with } \mathbf{ATR} = AT \cdot RT^t$$

$$\mathbf{d}_i^7 = \frac{\mathbf{ATK}(i,:)}{|T| \cdot |K|} \quad \text{with } \mathbf{ATK} = AT \cdot KT^t$$

To also include the negotiation needed by *i* for its *Tasks* into the measure we define the hamming distance between two matrices *X* and *Y* that are the same dimension $m \cdot n$ as the fraction of the cells that are different:

$$hd(X,Y) = \frac{^{m}_{i=1} \; _{j=1}(X(i,j) /= Y(i,j))}{m \cdot n}$$

And use this to calculate *Resourece* and *Knowledge* negotiation:

$$\mathbf{d}_i = \frac{hd(AR(i,:), [AT \cdot RT^t](i,:))}{|R|}$$

$$\mathbf{d}_i = \frac{hd(AK(i,:), [AT \cdot KT^t](i,:))}{|K|}$$

The last term represents number of *Agents* that *i* depends on or that depend on *i*. Let **s** be a vector describing the number of *Agents* that dependent on each *Task*

$$\mathbf{s} = (T + T^t) \cdot \quad AT(:,t)$$

DRAFT

And let **v** be the number of *Tasks* that *Agents* are dependent on

$$\mathbf{v} = AT \cdot s$$

Then

$$\mathbf{d}^{10} = \frac{v(i)}{|A| \cdot |T| \cdot (|T| - 1)}$$

Finally, the *Cognitive Demand* of *Agent i* is the average of all aspects:

$$\mathbf{d}^{i} = \frac{1}{10} \cdot \sum_{j=1}^{10} \mathbf{d}_j^{i}$$

Level: Node
Reference: Carley (2002)
Current ORA measures: cognitiveDemand

C.4.2 Awareness

A dyadic cognitive concept is *Awareness*. *Share Situation Awareness* measures the degree to which an *Agents* are similar, based on social interaction, physical distance, and socio-demographic data. *Shared Situation Awareness* is composed of four different terms that need the following three *AA* matrices:

A	Agent x Agent interaction/communication matrix
P	Agent x Agent physical proximity matrix
S	Agent x Agent social demographic similarity matrix

Together with

e	Eigenvector centrality measure computed on A
G	Dyadic geodesics computed on A

Shared Situation Awareness is defined as

$$\mathbf{a}(ij) = \alpha \cdot \mathbf{e}(i) \cdot \mathbf{e}(j) + \beta \cdot \mathbf{P}(i,j) + \frac{\delta \cdot \mathbf{S}(i,j)}{\gamma \cdot \mathbf{G}(i,j)} + \mu \mathbf{A}(i,j) \cdot \mathbf{A}(j,i)$$

The constants α, β, etc., control the influence of the specific terms and are set to 1.0 by default. The *AA* matrices described above can be approximated with networks including other node classes. First, the interaction/communication matrix can be replaced by every *AA* matrix. Second, the physical proximity matrix can be replaced by either (select from top to bottom)

DRAFT

$AL \cdot AL^t$	in case AL exists, or
Similarity of $[AT AE]$	in case AT and/or AE exist.

Otherwise ignore any physical proximity calculation. Third, the social demographic similarity matrix can be replaced with

Similarity of $[AKAR]$ in case AK and/or AR exist.

Otherwise ignore the social demographic similarity calculation.

Level: Graham et al. (2004)
Reference: Node, Dyad
Current ORA measures: sharedSituationAwareness

DRAFT

Appendix D

Julius Caesar Data

DRAFT

D.1 Interactions of Agents

Agent x Agent - Act 1

Agent x Agent - Act 2

Agent x Agent - Act 3

Agent x Agent - Act 4

Agent x Agent - Act 5

D.2 Agents and Their Connections

Agent x Knowledge

Agent x Location

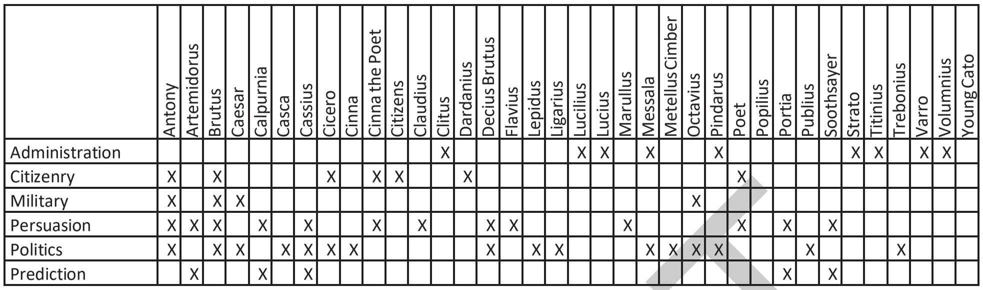
Agent x Event

Agent x Task

D.3 Other Networks

Event x Event

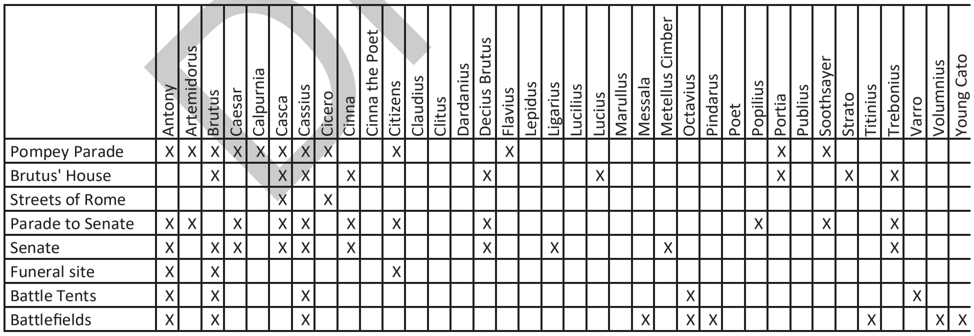
Knowledge x Task



	Antony	Artemidorus	Brutus	Caesar	Calpurnia	Casca	Cassius	Cicero	Cinna	Cinna the Poet	Citizens	Claudius	Clitus	Dardanius	Decius Brutus	Flavius	Lepidus	Ligarius	Lucilius	Lucius	Marullus	Messala	Metellus Cimber	Octavius	Pindarus	Poet	Popilius	Portia	Publius	Soothsayer	Strato	Titinius	Trebonius	Varro	Volumnius	Young Cato
Administration													X						X	X		X			X						X	X		X	X	
Citizenry	X		X					X		X	X			X												X										
Military	X		X	X																				X												
Persuasion	X	X	X		X		X			X		X			X	X					X					X		X		X						
Politics	X		X	X		X	X	X	X						X		X	X				X	X	X	X				X				X			
Prediction		X			X		X																					X		X						

Figure D.1: Agent by Knowledge link matrix

DRAFT



	Antony	Artemidorus	Brutus	Caesar	Calpurnia	Casca	Cassius	Cicero	Cinna	Cinna the Poet	Citizens	Claudius	Clitus	Dardanius	Decius Brutus	Flavius	Lepidus	Ligarius	Lucilius	Lucius	Marullus	Messala	Metellus Cimber	Octavius	Pindarus	Poet	Popilius	Portia	Publius	Soothsayer	Strato	Titinius	Trebonius	Varro	Volumnius	Young Cato
Pompey Parade	X	X	X	X	X	X	X	X			X					X												X		X						
Brutus' House			X			X	X		X						X					X								X			X		X			
Streets of Rome						X		X																												
Parade to Senate	X	X		X		X	X		X		X				X												X			X			X			
Senate	X		X	X		X	X		X						X			X					X										X			
Funeral site	X		X								X																									
Battle Tents	X		X				X																	X										X		
Battlefields	X		X				X															X		X	X							X			X	X

Figure D.2: Agent by Location link matrix

Location x Location

Task x Event

Task x Task

DRAFT

DRAFT

Bibliography

Adar, E. and Adamic, L. (2005). Tracking information epidemics in blogspace. In *Proceedings of the 2005 IEEE/WIC/ACM International Conference on Web Intelligence*, volume Compiegne, France.

Alderson, D. (2008). Catching the ”network science” bug: insight and opportunity for the operations researcher. *Operations Research*, 56:1047–1065.

Aldrich, H. E. and Herker, D. (1977). Boundary spanning roles and organization structure. *Academy of Management Review*, 2(2):217–230.

Allen, T. J. and Fustfeld, A. R. (1975). Research laboratory architecture and the structuring of communications. *R&D Management*, 5(2):153–164.

Anthonisse, J. M. (1971). The rush in a directed graph. Technical Report BN 9/71, Stichting Mathematisch Centrum, Amsterdam.

Ashworth, M. and Carley, K. M. (2003). Critical human capital. Unpublished Document: Carnegie Mellon University.

Ashworth, M. and Carley, K. M. (2006). Who you know vs. what you know: The impact of social position and knowledge on team performance. *Journal of Mathematical Sociology*, 30(1):43–75.

Axelrod, R. M. (1997a). *The Complexity of Cooperation: Agent-Based Models of Competition and Collaboration*. Princeton University Press, Princeton, NJ.

Axelrod, R. M. (1997b). *The Complexity of Cooperation: Agent-Based Models of Competition and Collaboration*. Princeton University Press.

Barabási, A.-L. (1999). Emergence of scaling in random networks. *Science*, 286:509–512.

DRAFT

Barabási, A.-L. and Albert, R. (1999). Emergence of scaling in random networks. *Science*, 286:509–512.

Barnes, J. A. (1954a). Class and committees in a Norwegian island parish. *Human Relations*, 7:39–58.

Barnes, J. A. (1954b). Class and committees in a Norwegian island parish. *Human Relations*, 7:39–58.

Barnlund, D. C. and Harland, C. (1963). Propinquity and prestige as determinants of communication networks. *Sociometry*, 26(4):467–479.

Behrman, R. Network analysis of the structure and capacity of brigade level military organizations. Unpublished Document: Carnegie Mellon University.

Berelson, B. (1952). *Content analysis in communication research*. Hafner, New York, NY.

Bernard, H. R. and Ryan, G. W. (1998). Text analysis: Qualitative and quantitative methods. In Bernard, H. R., editor, *Handbook of Methods in Cultural Anthropology*, pages 595–642. AltaMira, Walnut Creek, CA.

Berners-Lee, T., Hendler, J., and Lassila, O. (2001). The sematic web. *Scientific American*, 284(5):34–43.

Bonacich, P. (1972). Factoring and weighting approaches to status scores and clique identification. *Journal of Mathematical Sociology*, 2:113–120.

Bonacich, P. (2001). The evolution of exchange networks: A simulation study. *Journal of Social Structure*, 2(5).

Borgatti, S. and Everett, M. (1997). Network analysis of 2-mode data. *Social Networks*, 19(3):243–269.

Borgatti, S. P. (2009a). 2-mode concepts in social network analysis. In Meyers, R. A., editor, *Encyclopedia of Complexity and System Science*, pages 8279–8291. Springer-Verlag, New York, NY.

Borgatti, S. P. (2009b). 2-mode concepts in social network analysis. In Meyers, R. A., editor, *Encyclopedia of Complexity and System Science*, pages 8279–8291. Springer-Verlag, New York.

Brandes, U. (2001). A faster algorithm for betweenness centrality. *Journal of Mathematical Sociology*, 25(2):163–177.

DRAFT

Brandes, U. and Pich, C. (2007a). Eigensolver methods for progressive multidimensional scaling of large data. *Proceedings of the 14th International Symposium on Graph Drawing (GD'06)*, pages 42–53.

Brandes, U. and Pich, C. (2007b). Eigensolver methods for progressive multidimensional scaling of large data. *Proceedings of the 14th International Symposium on Graph Drawing (GD'06)*, pages 42–53.

Breiger, R., Carley, K. M., and Pattison, P., editors. *Dynamic Social Network Modeling and Analysis: Worskshop Summary and Papers*.

Breiger, R. L. (1974a). The duality of persons and groups. *Social Forces*, 53:181–90.

Breiger, R. L. (1974b). The duality of persons and groups. *Social Forces*, 53:181–90.

Burt, R. S. (1984). Network items and the general social survey. *Social Networks*, 6:293–340.

Burt, R. S. (1992a). *Structural Holes*. Cambridge University Press, Cambridge, MA.

Burt, R. S. (1992b). *Structural Holes*. Cambridge University Press.

Butts, C. and Carley, K. M. (2000). Spatial models of large-scale interpersonal networks.

Carley, K. M. (1988). Formalizing the social expert's knowledge. *Sociological Methods and Research*, 17(2):165–232.

Carley, K. M. (1990). Content analysis. In Asher, R.E., e. a., editor, *The Encyclopedia of Language and Linguistics*, volume 2, pages 725–730. Pergamon Press, Edinburgh, UK.

Carley, K. M. (1991). A theory of group stability. *American Sociological Review*, 56(3):331–354.

Carley, K. M. (1994). Extracting culture through textual analysis. *Poetics*, 22(4):291–312.

Carley, K. M. (1995). Communication technologies and their effect on cultural homegeneity, consensus, and the diffusion of new ideas. *Sociological Perspectives*, 38(4):547–571.

DRAFT

Carley, K. M. (1997). Network text analysis: The network position of concepts. In Roberts, C., editor, *Text Analysis for the Social Sciences: Methods for Drawing Statistical Inferences from Texts and Transcripts*, chapter 4, pages 79–100. Lawrence Erlbaum Associates, Inc., Mahwah, NJ.

Carley, K. M. (1999). On the evolution of social and organizational networks. *Research on the Sociology of Organizations*, 16:1–30.

Carley, K. M. (2001). Computational approaches to sociological theorizing. In Turner, J., editor, *Handbook of Sociological Theory*, pages 69–84. Kluwer Academic/Plenum Publishers, New York, NY.

Carley, K. M. (2002). Smart agents and organizations of the future. In Lievrouw, L. and Livingstone, S., editors, *The Handbook of New Media: Social Shaping and Consequences of ICTs*, pages 206–220. Sage, Thousand Oaks, CA.

Carley, K. M. (2003a). Computational organizational science and organizational engineering. *Simulation Modeling Practice and Theory*, 10(5–7):253–269.

Carley, K. M. (2003b). Dynamic network analysis. In Breiger, R., Carley, K. M., and Pattison, P., editors, *Dynamic Social Network Modeling and Analysis: Workshop Summary and Papers, Committee on Human Factors, National Research Council*, pages 133–145. National Research Council, Washington, DC.

Carley, K. M. (2004a). Dynamic social network modeling and analysis. In Breiger, R. L., Carley, K. M., and Pattison, P., editors, *2002 Workshop Summary and Papers*, pages 133–45. National Academies Press, Washington, DC.

Carley, K. M. (2004b). Estimating vulnerabilities in large covert networks using multi-level data. In *Proceedings of the NAACSOS 2004 Conference*, Pittsburgh, PA.

Carley, K. M. (2006). Destabilization of covert networks. *Computational and Mathematical Organization Theory*, 12(1):51–66.

Carley, K. M., Dekker, D., and Krackhardt, D. (2000). How do social networks affect organizational knowledge utilitization? Unpublished Document: Carnegie Mellon University.

DRAFT

Carley, K. M., Diesner, J., Reminga, J., and Tsvetovat, M. (2007). Toward an interoperable dynamic network analyssi toolkit, dss special issue on cyberinfrastructure for homeland security: Advances in information sharing, data mining, and collaboration systems. 43(4):1324–1347.

Carley, K. M., Fridsma, D., Casman, E., Yahja, A., Altman, N., Chen, L.-C., Kaminsky, B., and Nave, D. (2004). Biowar: Scalable agent-based model of bioattacks. *IEEE Transactions on Systems, Man and Cybernetics-Part A*, 36(2):252–265.

Carley, K. M. and Lee, J.-S. (1998). Dynamic organizations: Organizational adaptation in a changing environment. In Baum, J., editor, *Advanced in Strategic Management, Roots of Strategic Management Research*, volume 15, pages 269–297. JAI Press, Greenwich, CT.

Carley, K. M., Martin, M., and Hirshman, B. R. (2009a). The etiology of social change. *Topics in Cognitive Science*, 1(4):621–650.

Carley, K. M. and Palmquist, M. (1992). Extracting, representing and analyzing mental models. *Social Forces*, 70(3):601–636.

Carley, K. M., Pfeffer, J., Reminga, J., Storrick, J., and Columbus, D. (2012). ORA user's guide 2012. Technical Report CMU-ISR-12-105, Carnegie Mellon University, School of Computer Science, Institute for Software Research.

Carley, K. M., Reminga, J., and Borgatti, S. (2003). Destabilizing dynamic networks under conditions of uncertainty. In *IEEE KIMAS*, Boston, MA.

Carley, K. M., Reminga, J., Storrick, J., and De Reno, M. (2009b). Ora user's guide 2009. Technical Report CMU-ISR-09-115, Carnegie Mellon University, School of Computer Science, Institute for Software Research.

Carley, K. M. and Ren, Y. (2001). Tradeoffs between performance and adaptability for c3i architectures. *Proceedings of the 2001 Command and Control Research and Technology Symposium*.

Cartwright, D. and Harary, F. (1956). Structural balance: A generalization of Heider's theory. *Psychological Review*, 63:277–93.

Collins, A. and Quillian, M. (1969). Retrieval time from semantic memory. *Journal of Verbal Learning and Verbal Behavior*, 8(2):240–248.

Corman, S. R., Kuhn, T., McPhee, R. D., and Dooley, K. (2002). Studying complex discursive systems: Centering resonance analysis of organizational communication. *Human Communication Research*, 28(2):157–206.

DRAFT

Crane, D. (1972). *Invisible colleges. Diffusion of knowledge in scientific communities*. The University of Chicago Press, Chicago, IL.

Danowski, J. A. (1993). Network analysis of message content. In Richards, W. D. and Barnett, G. A., editors, *Progress in Communication Sciences*, volume 12. Norwood, NJ.

Davis, A., Gardner, B. B., and Gardner, M. R. (1941a). *Deep South: A Social Anthropological Study of Caste and Class*. University of Chicago Press, Chicago, IL.

Davis, A., Gardner, B. B., and Gardner, M. R. (1941b). *Deep South: A Social Anthropological Study of Caste and Class*. University of Chicago Press, Chicago.

Davis, G. and Carley, K. M. (2007). Computational analysis of merchant marine global positioning data. Technical Report CMU-ISR-07-109, Carnegie Mellon University, School of Computer Science, Institute for Software Research.

Davis, G. and Carley, K. M. (2008). Clearing the fog: Fuzzy, overlapping groups for social networks. *Social Networks*, 30(3):201–212.

Deerwester, S., Dumais, S., Furnas, G., Landauer, T., and Harshman, R. (1990). Indexing by latent semantic analysis. *Journal of the American society for information science*, 41(6):391–407.

Dekker, D., Krackhardt, D., and Snijders, T. A. B. (2007). Sensitivity of MRQAP tests to collinearity and autocorrelation conditions. *Psychometrika*, 72(4):563–581.

Diesner, J. and Carley, K. M. (2004). Automap 1.2 : extract, analyze, represent, and compare mental models from texts. Technical Report CMU-ISRI-04-100, Carnegie Mellon University, School of Computer Science, Institute for Software Research International.

Diesner, J. and Carley, K. M. (2005). Revealing social structure from texts: Meta-matrix text analysis as a novel method for network text analysis. In Narayanan, V. and Armstrong, D., editors, *Causal Mapping for Information Systems and Technology Research: Approaches, Advances, and Illustrations*, chapter 4, pages 81–108. Idea Group Publishing, Harrisburg, PA.

Diesner, J. and Carley, K. M. (2008). Conditional random fields for entity extraction and ontological text coding. *Journal of Computational and Mathematical Organization Theory*, 14:248–262.

DRAFT

Doerfel, M. (1998). What constitutes semantic network analysis? a comparison of research and methodologies. *Connections*, 21(2):16–26.

Dunbar, R. I. M. (1998). The social brain hypothesis. *Evol. Anthropol*, 6:178–190.

Epstein, J. and Axtell, R. (1996). *Growing Artificial Societies*. MIT Press, Boston, MA.

Faust, K. Centrality in affiliation networks. *Social Networks*, 19(3 , year =).

Faust, K. Centrality in affiliation networks. *Social Networks*, 19(3 , year =).

Festinger, L., Schachter, S., and Back, K. (1950). The spatial ecology of group formation. In Festinger, L., Schachter, S., and Back, K., editors, *Social Pressure in Informal Groups*, page Chapter 4. MIT Press, Cambridge, MA.

Freeman, L. C. (1977). A set of measures of centrality based on betweenness. *Sociometry*, 40:35–41.

Freeman, L. C. (1979). Centrality in social networks: Conceptual clarification. *Social Networks*, 1(3):215–239.

Freeman, L. C. (2000). Visualizing social networks. *Journal of Social Structure*, 1(1).

Fruchterman, T. M. J. and Reingold, E. M. (1991). Graph drawing by force-directed placement. *Software: Practice and Experience*, 21(11):1129–1164.

Gerner, D., Schrodt, P., Francisco, R., and Weddle, J. (1994). Machine coding of event data using regional and international sources. *International Studies Quarterly*, 38(1):91–119.

Gini, C. (1921). Measurement of inequality of incomes. *The Economic Journal*, 31:124–126.

Giuffre, K. (2001). Mental maps: Social netowrks and the language of critical reviews. *Sociological Inquiry*, 71(3):381–393.

Gladwell, M. (2000). *The Tipping point: how little things can make a big difference*. Little Brown, Inc, New York, NY.

Glaser, B. and Strauss, A. (1967). *The Discovery of Grounded Theory: Strategies for Qualitative Research*. Aldine de Gruyter, Hawthorne, NY.

DRAFT

Graham, J. M., Schneider, M., and Gonzalez, C. (2004). Report social network analysis of unit of action battle laboratory simulations. Technical Report CMU-SDS-DDML-04-01, Carnegie Mellon University, Social and Decision Sciences.

Granovetter, M. S. (1973). The strength of weak ties. *American Journal of Sociology*, 78(6):1360–1680.

Granovetter, M. S. (1974). *Getting a job*. University of Chicago Press, Chicago, IL.

Hamill, L. and Gilbert, N. (2009). Social circles: A simple structure for agent-based social network models. *Journal of Artificial Societies and Social Simulation*, 12.

Harrison, D., Price, K., and Bell, M. (1998). Beyond relational demography: Time and the effects of surface-and deep-level diversity on work group cohesion. *Academy of Management Journal*.

Heider, F. (1946). Attitudes and cognitive organizations. *Journal of Psychology*, 21:107–112.

Hill, R. A. and Dunbar, R. I. M. (2003). Social network size in humans. *Hum. Nature*, 14:53–72.

Hirschman, A. O. (1945). *National Power and the Structure of Foreign Trade*. University of California Press, Berkeley, CA.

Hirshman, B. R., Martin, M. K., and Carley, K. M. (2008). Modeling information access in construct. Technical Report CMU-CS-08-115, Carnegie Mellon University.

Hirshman, B. R. and St. Charles, J. (2009). Simulating emergent multi-tierd social ties. In *Proceedings of the 2009 Human Behavior and Computational Intelligence Modeling Conference*, volume Oak Ridge National Laboratory, TN.

Holland, P. W. and Leinhardt, S. (1976). The statistical analysis of local structure in social networks. In Heise, D. R., editor, *Sociological Methodology*, pages 1–45. Jossey-Bass, San Francisco, CA.

Hubert, L. J. and Schultz, J. (1976). Quadratic assignment as a general data analysis strategy. *British Journal of Mathematical and Statistical Psychology*, 29:190–241.

DRAFT

Janas, J. and Schwind, C. (1979). Extensional semantic networks. In Findler, N., editor, *Associative Networks. Representation and Use of Knowledge by Computers*, pages 267–302. Academic Press, New York, NY.

Johnson, J., Borgatti, S., Luczkovich, J., and Everett, M. (2001). Network role analysis in the study of food webs: An application of regular role coloration. *The Journal of Social Structure*, 2(3):1–15.

Kamada, T. and Kawai, S. (1989). An algorithm for drawing general undirected graphs. *Information Processing Letters*, 31:7–15.

Kamp, H. (1981). A theory of truth and semantic representation formal methods in the study of language. In Groenendijk, J., Janssen, T. M. V., and Stokhof, M. B. J., editors, *Formal Meithods in the Study of Language*, pages 277–322. Mathematical Centre Tracts 135, Amsterdam, Netherlands.

Kapferer, B. (1972). *Strategy and Transaction in an African Factory: African Workers and Indian Management in a Zambian Town*. University of Manchester Press, Manchester, England.

Karinthy, F. (1929). Chains. In Karinthy, F., editor, *Everything is Different*, page Chapter 4. Atheneum. Press, Budapest, Hungaria.

Katz, L. and Powell, J. H. (1955). Measurement of the tendency toward reciprocation of choice. *Sociometry*, 18(4):403–409.

King, G. and Lowe, W. (2003). An automated information extraction tool for international conflict data with performance as good as human coders: A rare events evaluation design. *International Organization*, 57(3):617–642.

Klimoski, R. and Mohammed, S. (1994). Team mental model: Construct or metaphor? *Journal of Management*, 20(2):403–437.

Krackhardt, D. (1987a). AQP partialling as a test of spuriousness. *Social Networks*, 9:171–186.

Krackhardt, D. (1987b). QAP partialling as a test of spuriousness. *Social Networks*, 9:171–186.

Krackhardt, D. (1988). Predicting with networks: Nonparametric multiple regression analysis of dynamic data. *Social Networks*, 10:359–381.

Krackhardt, D. and Carley, K. M. (1998). A pcans model of structure in organization. In *Proceedings of the International Symposium on Command and Control Research and Technology*, volume Monterray, CA, pages 113–119.

DRAFT

Lee, J.-S. and Carley, K. M. (2004). Orgahead: A computational model of organizational learning and decision making. Technical Report CMU-ISRI-04-117, Carnegie Mellon University, Social and Decision Sciences.

Lewis, E., Carley, K. M., and Diesner, J. (2003). Concept networks in organizational language: Consensus or creativity? In *Proceedings of the XXIII Sunbelt Social Network Conference*, volume Cancun, Mexico.

Lievrouw, L. A., Rogers, E. M., Lowe, C. U., and Nadel, E. (1987). Triangulation as a research strategy for identifying invisible colleges among biomedical scientists. *Social Networks*, 9(3):217–248.

Lin, N. (2001a). *Social Capital: A Theory of Social Structure and Action*. Cambridge University Press, Cambridge, MA.

Lin, N. (2001b). *Social Capital: A Theory of Social Structure and Action*. Cambridge University Press, Cambridge.

Magurran, A. E. (2003). *Measuring Biological Diversity*. Wiley-Blackwell, Malden, MA.

McCulloh, I. (2009). *Detecting Changes in a Dynamic Social Network*. PhD thesis, Carnegie Mellon University, School of Computer Science, Institute for Software Research.

McCulloh, I. and Carley, K. M. (2011). Detecting change in longitudinal social networks. *Journal of Social Structure*, 3:1–37.

McPherson, M., Smith-Lovin, L., and Cook, J. (2001). Birds of a feather: Homophily in social networks. *Annual Review of Sociology*, 27(1):415–444.

Milgram, S. (1967a). The small world problem. *Psychology Today*, 5:60–67.

Milgram, S. (1967b). The small world problem. *Psychology Today*, 5:60–67.

Milroy, J. and Milroy, L. (1985). Linguistic change, social network and speaker innovation. *Journal of Linguistics*, 21(2):339–384.

Mohr, J. (1998). Measuring meaning structures. *Annual Reviews in Sociology*, 24(1):345–370.

Monge, P. R. and Contractor, N. (2003). *Theories of Communication Networks*. Oxford University Press, New York, NY.

DRAFT

Moon, I. and Carley, K. M. (2006). Estimating the near-term changes of an organization with simulations. In *AAMAS*, volume Honolulu, HI, pages 111–118.

Moreno, J. L. (1953). *Who Shall Survive? Foundations of Sociometry, Group Psychotherapy and Sociodrama*. Beacon House, New York, NY.

Newcomb, T. (1961). *The Acquaintance Process*. Holt, Rinehart, and Winston, New York, NY.

Newman, M. (2003). The structure and function of complex networks. *SIAM Review*, 45(2):167–256.

Newman, M., Barabsi, A.-L., and Watts, D. J. (2006). *The Structure and Dynamics of Networks*. Princeton University Press, Princeton, NJ.

Page, E. (1961). Cumulative sum control charts. *Technometrics*, 3:1–9.

Pearl, J. (1988). *Probabilistic reasoning in intelligent systems: networks of plausible inference*. Morgan Kaufmann, San Francisco, CA, 1st edition edition.

Pearson, K. (1920). Notes on the history of correlations. *Biometrika*, 13:25–45.

Roberts, S. G. B., Dunbar, R. I. M., Pollet, T. V., and Kuppens, T. (2009). Exploring variation in active network size: Constraints and ego characteristics. *Social Networks*, 31:138–146.

Robins, G., Pattison, P. E., Kalish, Y., and Lusher, D. (2007a). An introduction to exponential random graph (p^*) models for social networks. *Social Networks*, 29(2):173–191.

Robins, G., Pattison, P. E., Kalish, Y., and Lusher, D. (2007b). An introduction to exponential random graph (p^*) models for social networks. *Social Networks*, 29(2):173–191.

Rogers, E. M. (2003a). *Diffusion of Innovations*. Free Press, New York, NY, 5th edition.

Rogers, E. M. (2003b). *Diffusion of Innovations*. Free Press, New York, 5th edition.

Rouse, W. and Morris, N. (1986). On looking into the black box; prospects and limits in the search for mental models. *Psychological Bulletin*, 100:349–363.

DRAFT

Ryan, G. W. and Bernard, H. R. (2000). Data management and analysis methods. In Denzin, N. K. and Lincoln, Y. S., editors, *Handbook of Qualitative Research*, pages 769–802. Sage Publications, Thousand Oaks, CA, 2 edition.

Sabidussi, G. (1966). The centrality index of a graph. *Psychometrika*, 31(4):581–603.

Sailer, K. and McCulloh, I. (2012). Social networks and spatial configuration—how office layouts drive social interaction. *Social Networks*, 34(1):47–58.

Sampson, S. F. (1969a). *Crisis in a cloister*. PhD thesis, Cornell University.

Sampson, S. F. (1969b). *Crisis in a cloister*. PhD thesis, Cornell University.

Schrodt, P., Yilmaz, O., Gerner, D. J., and Hermick, D. (2008). Coding substate actors using the cameo (conflict and mediation event observations) actor coding framework. In *Annual Meeting of the International Studies Association*, volume San Francisco, CA.

Shannon, C. E. (1948). A mathematical theory of communication. *The Bell System Technical Journal*, 27:379–423 and 623–656.

Shapiro, S. (1971). A net structure for semantic information storage, deduction and retrieval. In *Proceedings of the Second International Joint Conference on Artificial Intelligence*, pages 512–523. Barcelona, Spain.

Shewhart, W. (1927). Quality control. *Bell Systems Technical Journal*, 6(4):722–735.

Simon, H. (1957). A behavioral model of rational choice. In *Models of Man: Mathematical Essays on Rational Human Behavior in a Social Setting*, pages 241–60. John Wiley and Sons, ltd, London, England.

Sowa, J. (1992). Semantic networks. In Shapiro, S., editor, *Encyclopedia of Artificial Intelligence*, pages 1493–1511. Wiley and Sons, New York, NY, 2nd edition.

Sowa, J. F. (1984). *Conceptual Structures: Information Processing in Mind and Machine*. The Systems Programming Series. Addison-Wesley Publishing Company, Inc, Reading, MA.

Steglich, C., Snijders, T. A. B., and West, P. (2006). Applying sienna: an illustrative analysis of the co-evolution of adolescents' friendship networks, taste in music, and alcohol consumption. *Methodology*, 2(1):48–56.

DRAFT

Torgerson, W. S. (1952). Multidimensional scaling. *Psychometrika*, 17:401–419.

Travers, J. and Milgram, S. (1969). An experimental study of the small world problem. *Sociometry*, 32(4):425–443.

Van Atteveldt, W. (2008). *Semantic Network Analysis: Techniques for Extracting, Representing, and Querying Media Content*. BookSurge Publishing, Charleston, SC.

van Cuilenburg, J., Kleinnijenhuis, J., and de Ridder, J. (1986). A theory of evaluative discourse: Towards a graph theory of journalistic texts. *European Journal of Communication*, 1(1):65–96.

Wasserman, S. (1980). Analyzing social networks as stochastic processes. *Journal of American Statistical Association*, 75:280–294.

Wasserman, S. and Faust, K. (1994). *Social Network Analysis: Methods and Applications*. Cambridge University Press, Cambridge, MA.

Watts, D. J. (1999). Networks, dynamics, and the small world phenomenon. *American Journal of Sociology*, 105:493–527.

Watts, D. J. and Strogatz, S. H. (1998). Collective dynamics of 'small-world' networks. *Nature*, 393:440–442.

Wegner, D. (1986). Transactive memory: A contemporary analysis of the group mind. In Mullen, B. and Goethals, G. R., editors, *Theories of group behavior*, pages 185–208. Springer-Verlag, New York, NY.

Wellman, B. (1988). Structural analysis: From method and metaphor to theory and substance. In Wellman, B. and Berkowitz, S. D., editors, *Social Structure*, pages 19–61. Cambridge University Press, Cambridge, MA.

Woods, W. (1975). What's in a link: Foundations for semantic networks. In Collins, D. B. and A., editors, *Representationi and Understanding: Studies in Cognitive Science*, pages 35–82. Academic Press, New York, NY.

Zhou, W. X., Sornette, D., Hill, R. A., and Dunbar, R. I. M. (2005). Discrete hierarchical organization of social group sizes. *Proceedings of the Royal Society Biological Sciences*, 272(1561):439–444.

DRAFT