VISUALIZING REAL-WORLD NETWORKS

by

Lyndsay Roach,

B.A., Concordia University, 2015

A thesis presented to Ryerson University

in partial fulfillment of the

requirements for the degree of

Master of Science

in the program of

Applied Mathematics

Toronto, Ontario, Canada, 2018

AUTHOR'S DECLARATION

I hereby declare that I am the sole author of this thesis. This is a true copy of the thesis, including any required final revisions, as accepted by my examiners.

I authorize Ryerson University to lend this thesis to other institutions or individuals for the purpose of scholarly research.

I further authorize Ryerson University to reproduce this thesis by photocopying or by other means, in total or in part, at the request of other institutions or individuals for the purpose of scholarly research.

I understand that my thesis may be made electronically available to the public.

Visualizing Real-World Networks

Master of Science, 2018

Lyndsay Roach

Applied Mathematics

Ryerson University

Abstract

The study of networks has been propelled by improvements in computing power, enabling our ability to mine and store large amounts of network data. Moreover, the ubiquity of the internet has afforded us access to records of interactions that have previously been invisible. We are now able to study complex networks with anywhere from hundreds to billions of nodes; however, it is difficult to visualize large networks in a meaningful way.

We explore the process of visualizing real-world networks. We first discuss the properties of complex networks and the mechanisms used in the network visualizing software Gephi. Then we provide examples of voting, trade, and linguistic networks using data extracted from on-line sources. We investigate the impact of hidden community structures on the analysis of these real-world networks.

Acknowledgements

I am thankful to the Department of Mathematics at Ryerson University for a memorable experience during my master′s degree. I would like to give a special thank you Dr. Anthony Bonato for constantly going above and beyond in his responsibilities as my supervisor. I would also like to thank Dr. Peter Danziger and Dr. Dejan Delić for taking the time to be a part of my thesis committee.

Thank you to my family and friends for their endless support and encouragement throughout my education.

Table of Contents

Declaration ii

Abstract iii

Acknowledgements iv

Limitations of Inference v

List of Figures viii

1 Introduction 1

1.1 Motivation and Background 1

1.2 Basic Graph Theory Concepts 3

1.3 Properties of Complex Networks 8

1.4 Thesis Outline 13

2 Network Visualization Tools 15

2.1 Introduction 15

2.2 Graph Drawing 16

2.3 Gephi 18

2.3.1 ForceAtlas2 18

3 Canadian Political Networks 25

3.1 Canada’s House of Commons 26

3.1.1 Data Set 26

3.1.2 Methods 27

3.1.3 Results 28

3.1.4 Discussion 40

3.2 Ford Nation 43

3.2.1 Data Set 43

3.2.2 Methods 44

3.2.3 Results 44

3.2.4 Discussion 47

4 International Political Networks 50

4.1 Arms for Oil 51

4.1.1 Data Set 51

4.1.2 Methods 52

4.1.3 Results 53

4.1.4 Discussion 58

4.2 The Real Donald Trump 62

4.2.1 Data Set 62

4.2.2 Methods 62

4.2.3 Results 64

4.2.4 Discussion 69

5 Conclusion 73

5.1 Summary of Thesis 73

5.2 Future Work 74

Appendix 76

References 87

List of Figures

1 Graph $G(V,E)$, with $V(G) = \{a,b,c,d,e,f\}$, $E(G) = \{bc,ad,cd,ef,ac,cf,de\}$. 4

2 A directed graph G. 5

3 Graph H is a subgraph of graph G. 5

4 Graph H is an induced subgraph of graph G. 6

5 Graph H is a spanning subgraph of graph G. 6

6 Graph G and H are isomorphic with $f(a) = 1$, $f(b) = 6$, $f(c) = 8$, $f(d) = 3$, $f(g) = 5$, $f(h) = 2$, $f(i) = 4$, and $f(j) = 7$. 7

7 Graph G is a complete graph K_5 and graph H is a 3-regular graph. 8

8 Image (a) is a drawing of Konigsberg and image (b) is its graphical representation [8]. 15

9 Fruchterman and Reignold algorithm in Gephi. The first graph does not show communities and the second image shows communities. 17

10 The first graph is the basic ForceAtlas2 algorithm in Gephi and the second image is the ForceAtlas2 algorithm with the LinLog setting. 20

11 Examples of when the gravity setting is 2.0, and 6.0. 21

12 The ForceAtlas2 algorithm with the stronger gravity setting. 21

13 Example of when the scaling set to 3.0, and 8.0. 22

14 The first graph is without the Dissuade Hubs mode and the second graph is with the Dissuade Hubs mode. 23

15 The ForceAtlas2 algorithm with the prevent overlap mode. 24

16 Modularity partitioning of N_1. 29

17 Clustering coefficient partitioning of N_1. 30

18 Betweenness centrality scores of N_1. 31

19 Closeness centrality scores of N_1. 32

20 Modularity partitioning of N_2. 33

21 Clustering coefficient partitioning of N_2. 34

22 Betweenness centrality scores of N_2. 35

23 Closeness centrality scores of N_2. 36

24 Modularity partitions of N_3. 37

25 Clustering coefficient partitioning of N_3. 37

26 Betweenness centrality scores of N_3. 39

27 Closeness centrality scores of N_3. 40

28 Partitioning based on modularity score. 45

29 Partitioning based on clustering coefficients. 46

30 Dynamic graph of city councillors voting with Rob Ford over twenty votes. 47

31 Network N_{a1} of arms trade with a value of one hundred million TIV or more. 54

32 Network N_{a2} which is a subgraph of N_{a1} with only the African countries and top fifteen influential countries. 55

33 Network N_p of oil exported from African countries. 56

34 Planar graph, N_{map1}, with the node colouring of N_{a1} 57

35 Planar graph, N_{map2}, with the node colouring of N_p 58

36 Subgraph of N_p showing Gabon and the countries it exports oil to 61

37 Network N_A of top one hundred used words from the last quarter of 2015. . 65

38 Network N_B of top one hundred used words from the last quarter of 2016. . 66

39 Network N_C of top one hundred used words from the last quarter of 2017. . 67

40 Network N_D of top one hundred used words from all tweets in 2018. 68

1 Introduction

1.1 Motivation and Background

Mathematician Leonard Euler is considered to be the first to write an article using early graph theory [5]. In 1736, Euler wrote an article discussing what is known as the *problem of the Konigsberg bridges* [5]. Konigsberg was a city in Eastern Prussia separated by the River Pregel and seven bridges [5]. The problem involved planning a route around the city where one crosses each bridge only once [5]. Euler was the first to tackle this as a mathematical problem and in turn, developed ideas that would form basic concepts in graph theory [5]. Subsequently, graph theory has been used to study various types of physical, biological and social systems.

A graph may interpreted as a visual representation of distinct elements of a population and their connections, illustrated by dots and line respectively. Therefore, graphs lend themselves naturally to modelling these systems as networks and make it easier to visualize a system. Graph theory has proven to be a very effective tool for analyzing real-world networks and has led to the growing interdisciplinary field of *network science* [13, 26].

The reach and capabilities of network science has been propelled by the increase in availability of data and improvements in computing power [26]. This progress in technology now allows us to mine and analyze large graphs with anywhere from hundred to billions of elements. These large networks are called *complex networks*. Common examples of a complex network are the World Wide Web graph of page links, the Hollywood graph of co-stars and the collaborations graph of coauthors [7]. While large, complex networks are made up of a discrete number of objects that evolve over time [13]. Complex networks are difficult to visualize, so understanding the dynamics and hidden community structure of a network are powerful aspects of visualizing real-world networks [7, 13, 26].

In this thesis, we will be discussing different types of networks and the properties of their graphical representations. The graphs that we will be examining have orders in the hundreds and include topics such as voting patterns, trading patterns, and linguistic patterns. There are three main characteristics of complex networks that are widely studied: low average distances between members in the network, high local densities, and the distribution of links between members [13]. We will be discussing these characteristics in more detail in the next section.

Human social systems do not organize the same way as physical systems [31]. We imagine physical systems in 3-dimensional Euclidean space, where coordinates correspond to a commonly known definition of Euclidean distance. One of the axioms of Euclidean distance is the *triangle inequality*, which defines a transitive relationship between three elements in Euclidean space. Interestingly, social systems violate this property [31]. This means that person A may know person B and person C, but person B and person C do not necessarily know each other [31]. Usually members of a particular social group are more likely to be acquainted with those within the group than they are to those in other social circles; however, we presumably belong to many different groups [26, 31]. For example, person A, person B and person C all go to different high schools in the same city. Person A attends gymnastics with person B and lives on the same street as person C; therefore, person A is friends on Facebook with both person B and person C, but person B and person C are not because they are not acquainted with one another.

Links between members of a network are often not related to physical distance, but are related by common interests. Therefore, we can think of a network as being embedded in a "social space" [31]. A social space refers to the idea that elements of a network, typically refering to people, that are "close" in a network share commonalities. The position of members in a network represent quantitative measures of their shared attributes [31]. Graphs are a practical and effective way of visualizing real-world networks in a social space, making it easier to analyze the information.

1.2 Basic Graph Theory Concepts

A network can be modelled by a *graph* $G(V, E)$, where $V(G) = V$ is a nonempty set of *nodes*, also called *vertices*, and $E(G) = E$ is a set of *edges*. The nodes represent the elements in the network and the edges are pairs of nodes representing predefined links between the elements [13, 31]. The *order* of a graph G is the number of vertices in $V(G)$, expressed as $|V(G)| = n$, and the *size* of a graph G is the number of edge in $E(G)$, expressed as $|E(G)| = m$ [31]. Each edge is a pair of nodes $e_k = \{v_i, v_j\}$ with $i \neq j$ and the nodes v_i and v_j are said to be *neighbours* or *adjacent* [7]. If there is an edge $e_k = \{v_i, v_j\}$ with $i = j$, then the edge is called a *loop*, meaning it starts and ends at the same node. If nodes v_i and v_j form an edge in graph G, then we shall use the notation v_iv_j to represent the edge $v_iv_j \in E(G)$. The edge $v_iv_j \in E(G)$ is *incident* to both v_i and to v_j [7]. In Figure 1 we have an example of a graph G with $|V(G)| = 6$ and $|E(G)| = 7$.

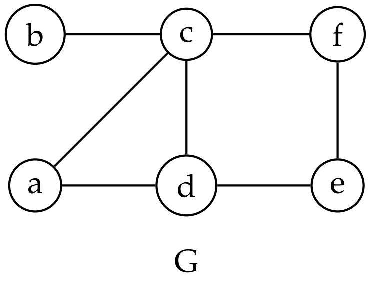


Figure 1: Graph $G(V, E)$, with $V(G) = \{a, b, c, d, e, f\}$, $E(G) = \{bc, ad, cd, ef, ac, cf, de\}$.

The number of neighbours a node has is called the *degree* of a node, denoted $\deg(v)$. The maximum degree of a graph G is denoted by $\Delta(G)$ and its minimum degree is denoted by $\delta(G)$ [13]. Loops are counted twice, so if $v_i v_i$ is the only edge from v_i, then $\deg(v_i) = 2$ [13]. If we consider graph G in Figure 1, we have for example $\deg(a) = 2$, $\Delta(G) = 4$, and $\delta(G) = 1$. A useful statistic to consider is the *average degree* of a graph, defined as [13]:

$$\deg_{av}(G) = \frac{1}{n} \sum_{v \in V(G)} \deg(v).$$

The *neighbourhood* of a node v is defined as the subgraph consisting of the nodes adjacent to v, but not including v, denoted $N(v)$ [31]. For example, in Figure 1, $N(f) = \{c, e\}$. A graph is called a *directed* graph or a *digraph* if the edges are ordered pairs of nodes, otherwise it is called *undirected* [7]. For the most part we will be discussing undirected graphs. Figure 2 shows an example of a directed graph. The order of the nodes connected by an edge in a directed graph is illustrated by an arrow.

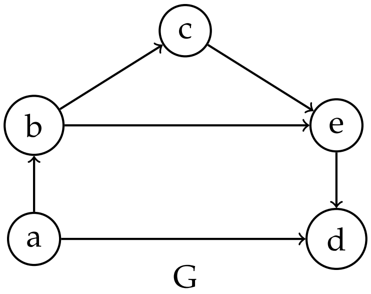


Figure 2: A directed graph G.

If a graph contains loops or multiple edges then it is a *multi-graph*, otherwise it is a *simple* graph [7]. Additionally, graphs can have edges that are assigned labels called *weights* [13]. The weights are real numbers whose value have a predetermined meaning; for example, the numbers of times that connection occurs, the importance of the connection, or the cost of the connection [13]. These are called *weighted graphs*.

A *subgraph* is a subset of nodes and edges of graph G [13]. An *induced subgraph* of G, referred to as a *module*, is a subgraph where the subset of nodes include the endpoints of all the edges in the subgraph [13]. A *spanning subgraph H* is a subgraph where $V(H) = V(G)$ [13].

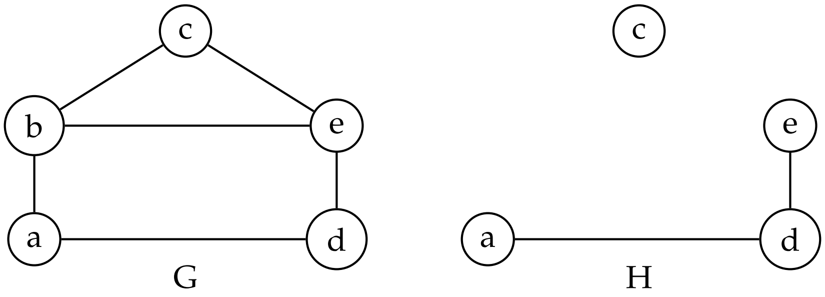


Figure 3: Graph H is a subgraph of graph G.

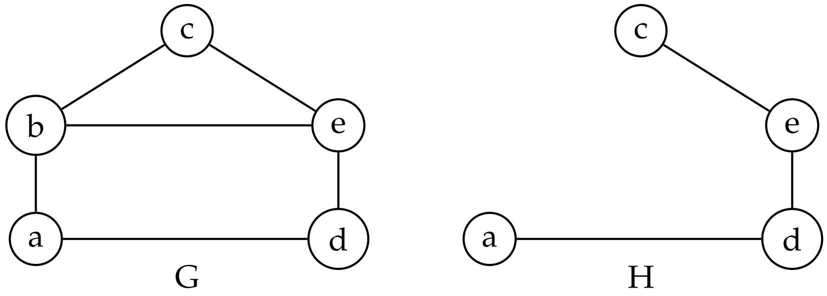


Figure 4: Graph H is an induced subgraph of graph G.

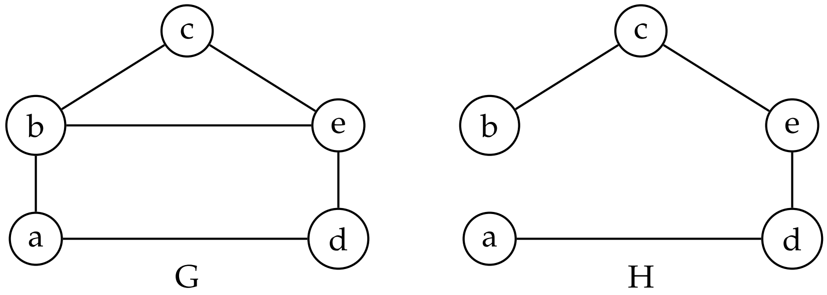


Figure 5: Graph H is a spanning subgraph of graph G.

A *path*, denoted P_n, is an ordered sequence of n nodes with $(n-1)$ edges [7]. In Figure 3 graph G, an example of a path P_5 of length 4 is $b \rightarrow e \rightarrow c \rightarrow b \rightarrow a$. A graph is *connected* if every distinct pair of nodes is connected by a path; otherwise, it is *disconnected* [13]. If a node v does not have any neighbours, meaning it has $\deg(v) = 0$, then it is called an *isolated node*. In Figure 3, graph H is an example of a disconnected graph and node c is an example of an isolated node. If a graph consists of connected induced subgraphs, then each connected induced subgraph is called a *connected component* [13].

Let G and H be graph and let $f : V(G) \rightarrow V(H)$ be a function. f is an *embedding* if it is injective, meaning $v_i v_j$ is an edge in G if and only if $f(v_i)f(v_j)$ is and edge in H [13]. f is an *isomorphism* if and only if it is a surjective embedding [13]. If f is an isomorphism, then G and H are called *isomorphic graphs*. Graph isomorphisms are important to keep in mind because when visualizing networks, how the network is presented can affect how it is interpreted. Figure 6 is an example of isomorphic graphs.

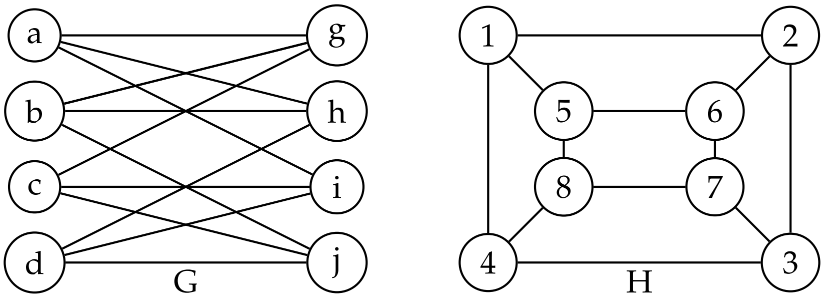
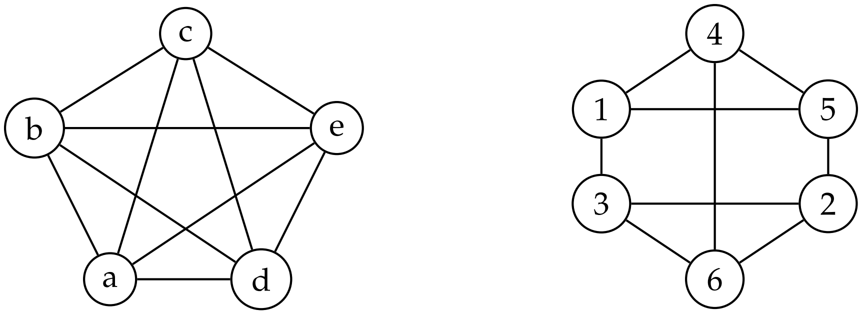


Figure 6: Graph G and H are isomorphic with $f(a) = 1$, $f(b) = 6$, $f(c) = 8$, $f(d) = 3$, $f(g) = 5$, $f(h) = 2$, $f(i) = 4$, and $f(j) = 7$.

A graph is called a *clique* or a *complete graph*, denoted K_n, if all n nodes are adjacent to each other [13]. The maximum size for a complete graph is $|E(G)| = \binom{n}{2} = \frac{n(n-1)}{2}$ and a graph is considered sparse if $|V(G)| \ll \frac{n(n-1)}{2}$ [31]. The density of a graph, denoted D_n, is the ratio of the number of edges in the graph and the number of edges in a complete graph of the same order; namely, $D_n = \frac{2|E(G_n)|}{n(n-1)}$ [13]. A *k-regular* graph is a graph where every node has degree k [13]. A *planar graph* is a graph that can be drawn in a plane without any edges crossing [7]. We will see an example in Chapter 4. Figure 7 gives an example of a complete graph and a k-regular graph.



G

H

Figure 7: Graph G is a complete graph K_5 and graph H is a 3-regular graph.

On an unweighted graph the distance between two nodes v_i and v_j, denoted $d(v_i, v_j)$, is the length of the shortest path connecting v_i and v_j [13]. Whereas, on a weighted graph, the distance between two nodes is the sum of the weights of the shortest path connecting the two nodes [13]. The maximum distance over all pairs of nodes on a graph G is called the *diameter* of G, denoted $\text{diam}(G)$ [13]. Also, one may want to consider the *average path length* of a graph which is the sum distances between all pairs of vertices divided by the total number of edges, $L(G) = \frac{\sum_{u,v \in V(G)} d(u,v)}{\binom{n}{2}}$ [32]. Note that the diameter is at least the length of the average path length [7].

1.3 Properties of Complex Networks

Complex networks are large networks with anywhere from hundreds to millions of nodes. The study of complex networks has progressed rapidly due to improvements in computing power [13]. Technological advancements have not only improved our ability to mine and use data, but it has presented more avenues for creating data. An important advancement in the study of complex networks, specifically social networks, is the increased accessibility to the internet. Social interactions occur increasingly on-line; therefore, there are now records of interactions that may not have been recorded before the internet. For example interactions between friends that transpire on Facebook or financial transactions from on-line shopping. The following section details three primary properties of social networks: power law degree distribution, small world phenomenon, and high clustering.

The first property we will discuss is the *power law distribution*. One of the features of a graph that can be analyzed is the frequency of degrees. Let n_k be the number of nodes with degree k, then the *degree distribution* of graph G is $(n_0, n_1, n_2, \ldots, n_t)$ where n_t is the largest degree [7]. The graph G has a power law distribution with exponent β if n_k is proportional to $k^{-\beta}$ for some fixed $\beta > 1$ [7]. This means that there are a low number of nodes with high degrees and many nodes with low degrees. Discussions about the power law distribution can be traced back to economist Wilfredo Pareto in 1896 [7]. He proposed that across countries and across time, the distribution of income and wealth followed this pattern [7]. A more current example is if you consider the network of Twitter users, where the users are the nodes and there is an edge if one user follows another user. There are a few users, typically celebrities, who have millions of followers compared to majority of users that have few followers. Studying the degree distribution is important because depending on the network in question, one can use to their advantage knowing the nodes with high degrees. For example, one can use it to spread information quickly or to stop the spread of a virus.

Another interesting property of social networks is what is called the *Small World phenomenon*. Informally, it describes the phenomenon when you encounter someone that you think you have no connection to, but then you find out that you have a mutual acquaintance. In 1967, psychologist Stanley Milgram took this idea and conducted an experiment to find how short these chains of mutual acquaintances tend to be [7]. He had 60 letters sent out to various participants in different parts of the United States and requested that they send it to a specific recipient in Cambridge, Massachusetts [7]. They could only pass the letter by hand to acquaintances either directly or through a friend of a friend [7]. Milgram concluded that the average length of these chains of acquaintances was between 5 and 6 [7]. Since Milgram's experiment, the Small World phenomenon has been observed in many different networks, not only in social networks [7]. Considering that most individuals are a part of many different social groups, the shortest paths between each group is important for utilising the structure of the network.

One can consider the interactions between groups, but one can also examine the relationships within a group. Social networks tend to have high clustering, meaning that the groups of nodes in the network are densely connected. This is because nodes are likely to share common neighbours [7]. The *local clustering coefficient*, denoted $C(v)$, is a valuable statistic when looking at the grouping of a network. It is the ratio of the number of edges in the neighbour of a node over the total number of possible edges in that neighbourhood, defined as:

$$C(v) = \frac{|E(N(v))|}{\binom{k_{n,v}}{2}},$$

where $k_{n,v}$ is the number of nodes in $N(v)$ [31]. In addition, there is the *global clustering coefficient* which is the average of the local clustering coefficients over all the nodes in the graph. It is defined as:

$$C(G) = \frac{1}{n} \sum_{v \in V(G)} C(v).$$

We may view the local clustering coefficient as a measurement of the extent to which a person's acquaintances are acquainted, or the probability that two people that have mutual acquaintances are acquainted themselves [31]. Clustering in a network shows which members are close together in the *social space* sense; thus, we assume that people that are neighbours in a network share characteristics. Measuring the clustering coefficients of a network may be useful to someone that is interested in giving their customers recommendations [26]. If customer A and customer B have similar purchase histories, then one may want to suggest to customer B a product that only customer A has purchased.

We can also look at the nuances of a network by considering the influence of each node in a cluster. The significance of a node is called the *centrality* of a node. The most straightforward way of measuring centrality is by the *degree centrality* of a node, meaning looking to the degree of the node to determine if it is important [13]. Let us recall the Twitter example. If millions of people follow a particular celebrity on Twitter, then a cluster forms around that person and they are an influential member of the network. If the network follows the power law degree distribution, there will be few of these high degree influential nodes.

The number of neighbours is not the only way one can measure the impact of a node on the network. One can also measure the *closeness centrality* by looking at the average distance a node is to the other nodes in the whole network [13], defined as:

$$C_c(v_i) = \frac{1}{\sum\limits_{v_{j \in V(G)}} d(v_i, v_j)}.$$

The number of shortest paths between two nodes v_s and v_t that pass through a node v_i is called the *stress centrality* of v_i, defined as:

$$C_s(v_i) = \sum_{s \neq t \neq i} \sigma_{st}(v_i),$$

where $\sigma_{st}(v)$ is the number of shortest paths between nodes v_s and v_t that pass through v_i [13]. Furthermore, there is *betweenness centrality* which is the ratio of the stress centrality over the total number of shortest paths between two nodes v_s and v_t, defined as:

$$C_b(v_i) = \sum_{s \neq t \neq i} \frac{\sigma_{st}(v_i)}{\sigma_{st}},$$

where σ_{st} is the total number of shortest paths between nodes v_s and v_t [13]. The betweenness centrality is interesting because it emphasises the importance of the nodes based on if they are connecting the different clusters in the network. Therefore, in this case a node can have a low degree but still be significant to the network because it is on the shortest path between two clusters [13]. These clustering statistics are important for community detection in a social network and understanding the underlying structure of the network [13].

Community detection not only plays a crucial role in understanding the structure of the network, it is also essential to network visualization [26]. Another statistic one may want to consider is the modularity, meaning the measure of the strength of community partitioning [13]. Let e_{st} be the edges that connect nodes in community s to the nodes in community t and let a_s be the the number of nodes that are adjacent to nodes in group s, specifically $a_s = \sum_t e_{st}$ [13, 26]. Modularity is defined as:

$$Q = \sum_{s=1}^{k} (e_{ss} - a_s^2).$$

As mentioned earlier, when networks have a high order and size it begins to get difficult to visualize the networks in a practical manner. The graphs that will be discussed in later chapters were created using visualization software called Gephi, which uses modularity to decide how to partition the communities. High modularity is viewed as positive because it means that there more edges in the community than expected and the correct partition was made [13, 26]. Community structure is detected by optimizing the partitions between clusters and maximizing the modularity; however, optimizing the modularity of network depends on its scale [26]. The *limit resolution* refers to the intrinsic scale of a network and communities beyond this natural scale may not be detected by modularity [26]. Moreover, the degree distribution of a network has notable effects on modularity [26]. If the degree distribution is too even, then it is hard to optimise the modularity of the network [26]. The effects of communities on network visualization using software will be discussed further in the next chapter.

1.4 Thesis Outline

This first chapter has described the motivation behind graphs and visualizing networks. It also outlined some basic definitions of graph theory and some basic concepts of social networks. Chapter 2 will detail force-directed layout algorithms and the ones used in the network visualizing software Gephi.

Chapter 3 will provide examples of graphs using data related to Canadian politics. It will include the methods used to create the graphs, along with an analysis and discussion of the results. Chapter 4 will give additional examples of graphs, but using data related to international politics. Finally, Chapter 5 will conclude the thesis with a summary of results and suggestions for future research.

2 Network Visualization Tools

2.1 Introduction

Using graphs to visualize real-world networks is key to studying the structure of a network. The problem of the Königsberg bridges is a good example of a graph being an effective way to representing a network. The Königsberg bridges graph had only 5 nodes and 7 edges, making it is simple to draw by hand as seen in Figure 8.

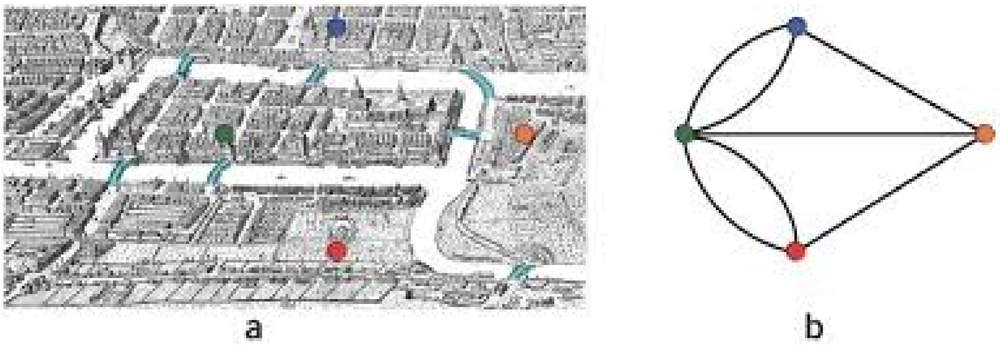


Figure 8: Image (a) is a drawing of Konigsberg and image (b) is its graphical representation [8].

We can imagine that graphs of higher orders get increasingly more difficult to draw by hand; hence, it would be impossible to draw useful representation of a complex network by hand. However, as mentioned in the previous chapter, visualizing complex networks is essential to understand their structures.

The increase in availability of data has made it necessary to develop better resources for visualizing large graphs [18]. There are many software tools for visualizing networks to choose from. The one used in this thesis is called Gephi. Gephi is a network visualizing software that can handle graphs with order in the tens of thousands of nodes and it is aimed at implementing various graph drawing tools, while staying user friendly [19, 20]. The last section of this chapter will provide details about visualizing networks specifically with Gephi.

2.2 Graph Drawing

The goal of drawing a graph is to assign each node a position in a low dimensional space, usually 2-dimensions, with the purpose of revealing their important connections [18]. Many real-world systems can be broken down into clusters, where there are dense connections within each cluster and sparse connections between the clusters [25]. These sub-groups should be clearly depicted in a graph visualization. Neighbouring nodes should be drawn close together in clusters and each cluster should be distinct. To obtain these results, one of the most common types of algorithms used for drawing network graphs is called force-directed layout algorithms [18]. Note that the following force-directed algorithms were developed with undirected graphs, drawn with straight edges in mind [14, 18]. However, at least in Gephi, the available algorithms are performed on all graphs.

In 1984, Eades was the first to introduce a force-directed layout using the spring-electrical model [18]. Eades' idea was to imagine the nodes as steel rings and the edges as springs, and then once released from its initial layout the spring forces would adjust the placement of the rings until the system reached a minimal energy state [14]. He used his own formula for the springs, rather than the conventional Hooke's law [14]. He calculated the repulsive forces between each pair of nodes and the attractive forces between neighbouring nodes [14]. In 1991, Fruchterman and Reingold built upon Eades' model with the analogy of rings and spring, but they developed new formulas [14]. Their concerns were that neighbouring nodes should be drawn close together, but nodes should not be drawn too close together [14]. They came up with an efficient algorithm using the attracting and repulsing forces, $F_a = \frac{d^2}{k}$ and $F_r = -\frac{d^2}{k}$ respectively, with k adjusting the scaling of the network [20].

The Fruchterman and Reingold algorithm is included in Gephi. All of the following examples are from the same random graph of order 100 generated in Gephi and each feature is demonstrated separately. Figure 9 is an example of the Fruchterman and Reingold algorithm in Gephi. Notice how the nodes are relatively spaced out and while it is visually appealing, clusters are hard to detect at first glance.

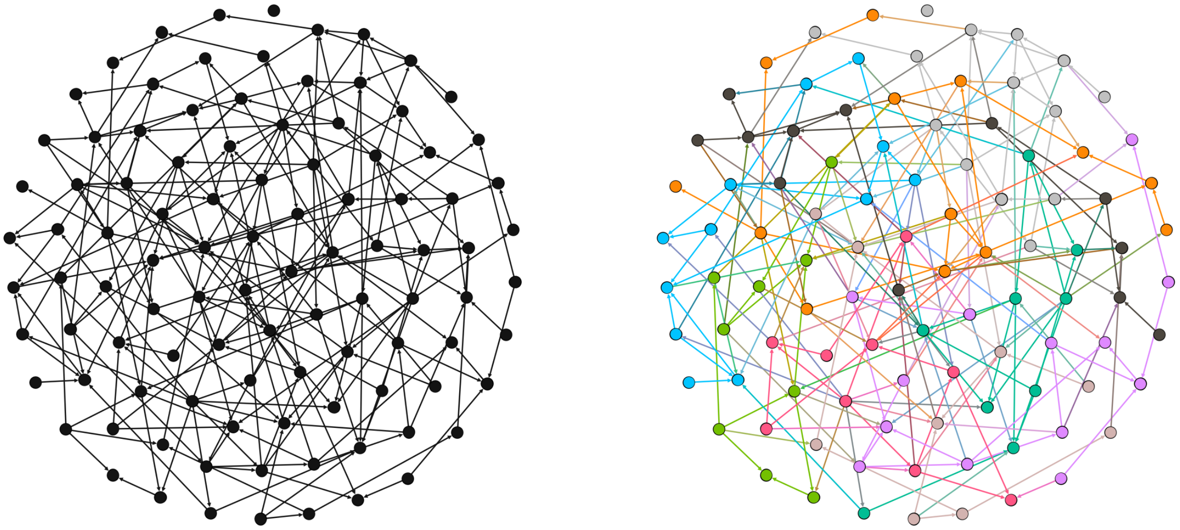


Figure 9: Fruchterman and Reignold algorithm in Gephi. The first graph does not show communities and the second image shows communities.

More recently, in 2007, Noack introduced his node-repulsion LinLog and edge-repulsion

LinLog versions of the force-directed algorithm. Noack points out that his LinLog algorithms show clusters more clearly than Fruchterman and Reingold's algorithm, especially the edge-repulsion algorithm [20]. LinLog is included as a setting for another algorithm in Gephi, but not as a separate option so we will look at an example later in the chapter.

2.3 Gephi

Gephi is an open source software for visualizing networks. It can handle large networks, up to the tens of hundreds of nodes [19]. The creators' main focus was creating a software that the user can manipulate in real-time [19]. Large graphs are difficult to lay out and interactive manipulation makes it easier for the user to obtain a meaningful visualization [18]. Users can import their own data and adjust the networks as they wish, using different filters and settings [19]. The creators intended Gephi for users with any level of background in graph theory [20]. The software includes many options of algorithms to choose from. Its default algorithm is called ForceAtlas2, which is an improved version of the original ForceAtlas developed by Gephi's creators and we will use it as an example to showcase some of Gephi's features [20].

2.3.1 ForceAtlas2

ForceAtlas2 is a continuous layout algorithm. It does not introduce anything new, but it combines effective features of previous force-directed algorithms [20]. ForceAtlas2 is strongly inspired by Noack's LinLog forces and even has a LinLog mode setting [20]. Like previous algorithms, ForceAtlas2 has all the nodes repulse each and the neighbouring nodes attract, simulating a spring-electrical model [20]. Since it is a continuous algorithm, the nodes will keep repulsing and attracting as long as it is running, but does converge to a balanced state [20]. ForceAtlas2 uses the classical attraction force $F_a(n_1, n_2) = d(n_1, n_2)$, where the attraction force depends linearly on the distance between n_1 and n_2 [20]. The formula used for the repulsion force [20] is

$$F_r(n_1, n_2) = k_r \frac{(\deg(n_1) + 1)(\deg(n_2) + 1)}{d(n_1, n_2)}.$$

They use $\mathrm{deg(n_i)} + 1$ to ensure that even the isolated nodes have a repulsive force [20]. The reason for having a degree-dependent repulsive force is because of the power law distribution property of many real-world networks. It results in nodes with low degrees being assigned positions closer to high degree nodes, making the graph more readable [20].

ForceAtlas2 offers different settings for the user to play with, so they can tailor the visualization to their needs. As previously mentioned, there is the *LinLog mode* that uses the attraction formula $F_a = \log(1 + d(n_1, n_2))$, which outputs better placements from a modularity standpoint [20]. Here is an example in Figure 10.

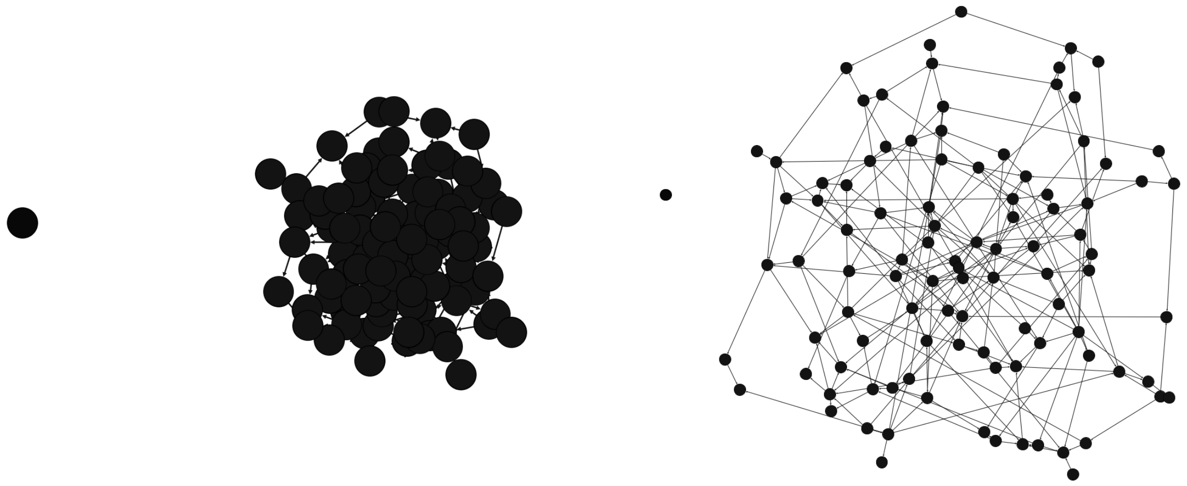


Figure 10: The first graph is the basic ForceAtlas2 algorithm in Gephi and the second image is the ForceAtlas2 algorithm with the LinLog setting.

There is also the *gravity* setting in Gephi, which prevents disconnected components from repulsing too far away from each other. It uses force

$$F_g(n) = k_g(\deg(n) + 1),$$

where k_g is set by the user [20].

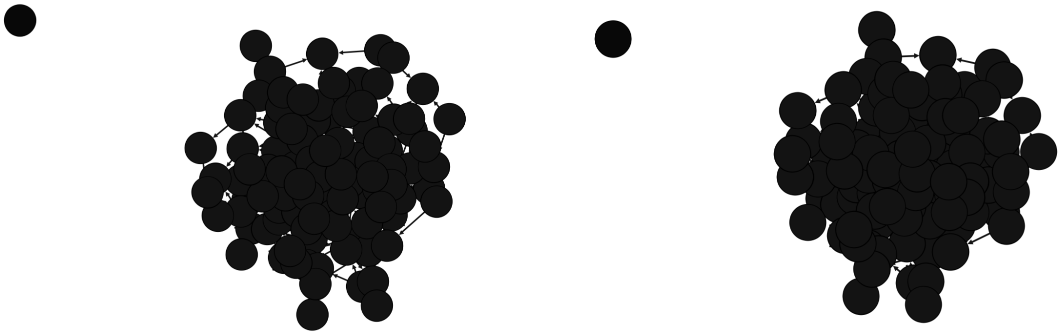


Figure 11: Examples of when the gravity setting is 2.0, and 6.0.

There is also the *stronger gravity* setting in Gephi which uses the force

$$F'_g(n) = k_g(\deg(n) + 1)d(n),$$

where $d(n)$ is the distance of the far away nodes from the centre [20]. The stronger gravity setting tends to not produce a readable graph, as seen in Figure 12.

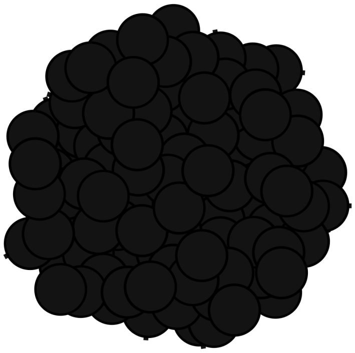


Figure 12: The ForceAtlas2 algorithm with the stronger gravity setting.

The user can also adjust the *scaling* option. ForceAtlas2 allows for the scale of k_r to be altered, but not k_a [20]. Increasing k_r expands the layout of the graph. Here are some examples of different scaling of k_r in Figure 13.

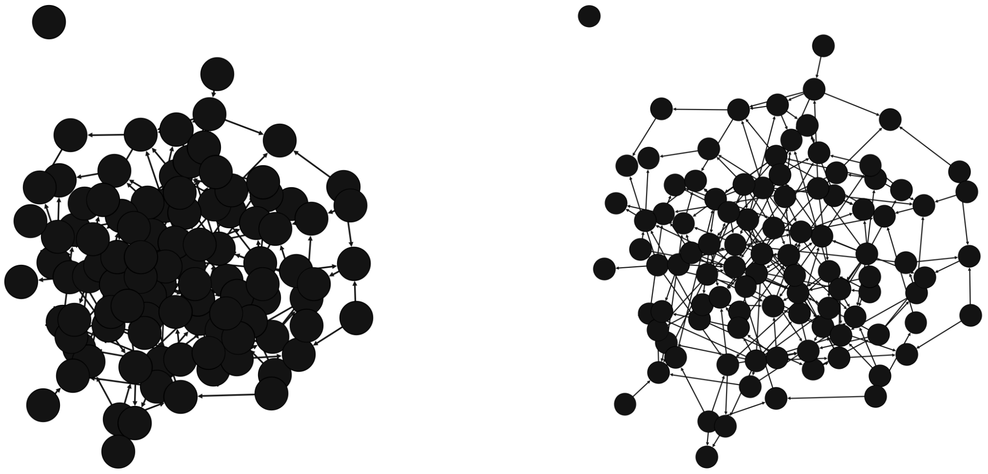


Figure 13: Example of when the scaling set to 3.0, and 8.0.

If the graph is weighted, then the weights affect the attraction force [20]. Then the attraction force formula becomes

$$F_a = w(e)^{\delta} d(n_1, n_2),$$

where $w(e)$ is the weight of edge e [20]. If $\delta = 0$, then the edge weights are ignored, and if $\delta = 1$, then the attraction force is proportional to the weight [20]. We do not have an example because our sample graph is unweighted.

Another option is the *Dissuade Hubs* mode. This setting is meant to keep nodes with high indegree closer to the centre and the nodes with high outdegree on the periphery [20]. The attraction force becomes

$$F_a(n_1, n_2) = \frac{d(n_1, n_2)}{\deg(n_1) + 1}.$$

Here is an example in Figure 14.

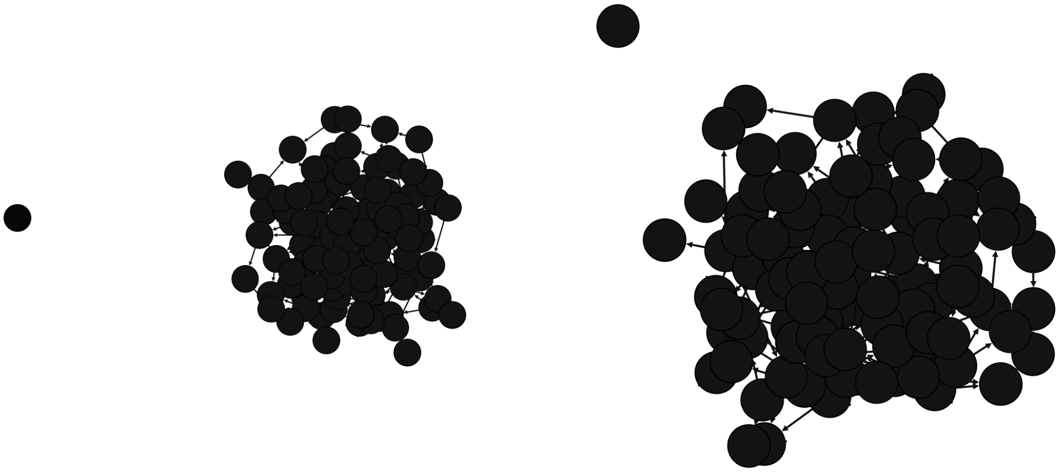


Figure 14: The first graph is without the Dissuade Hubs mode and the second graph is with the Dissuade Hubs mode.

Finally, there is the *prevent overlapping* feature. It considers the size of each node, $size(n)$, by computing $d(n_1, n_2)$ in both the attraction force and the repulsion force to layout the graph in a more readable manner [20].

In this case [20], Gephi uses

$$d'(n_1, n_2) = d(n_1, n_2) - size(n_1) - size(n_2).$$

If $d'(n_1, n_2) > 0$, then there is no overlapping and Gephi uses $d'(n_1, n_2)$ instead of $d(n_1, n_2)$ to compute $F_a(n_1, n_2)$ and $F_r(n_1, n_2)$ [20]. If

$d'(n_1, n_2) < 0$, then there is overlapping and $F_a(n_1, n_2) = 0$ and

$F_r(n_1, n_2) = k'_r(\deg(n_1) + 1)(\deg(n_2) + 1)$ [20]. However, if $d'(n_1, n_2) = 0$, then there is no attraction or repulsion [20]. The following figure is an example in Gephi using the no overlap setting.

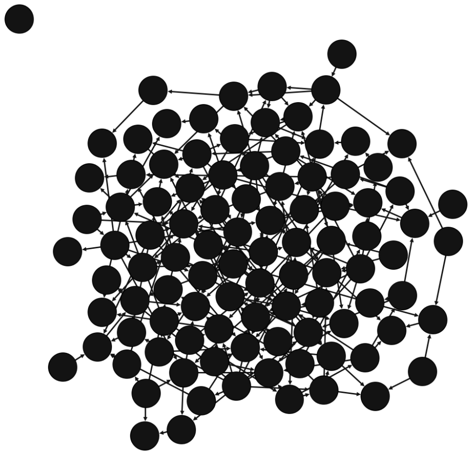


Figure 15: The ForceAtlas2 algorithm with the prevent overlap mode.

While all these examples demonstrated the features added to ForceAtlas2 separately, but any combination of these features is possible. Depending on the properties of the network, different settings will reveal different aspects of the network. In the next two chapters, we will look at examples of real-world networks and how we can utilize Gephi to extract meaningful conclusions.

3 Canadian Political Networks

One area in particular that network science has been proven valuable is in political science. For instance, graph theory has been used to analyze political networks representing members in the United States (U.S.) congress based on the committees and subcommittees they are apart of, on the bills they co-sponsor, and on the shared roll-call votes [24]. Furthermore, the idea of extracting a network of shared voting patterns has been applied to the Italian Parliament. The network was represented by a weighted graph where the nodes were the deputies and there is an edge between two deputies if they voted the same way on an issue [24]. For details, see reference [24]. In this chapter, we will explore two examples of political networks extracted from data relating to Canadian politics.

First, we will consider the voting patterns of Members of Parliament (MPs) during recent parliamentary sessions. Like most Westminster-style parliaments, Canadian MPs follow strict party discipline and tend to vote with their party [15]. However, there are interesting alliances between parties when it comes to trying to pass a bill or prevent a bill from being passed.

Secondly, we will consider the voting patterns of Toronto city councillors while Rob Ford was mayor. From 2010-2014, Rob Ford was mayor of Canada's largest city, Toronto. He had a very strong public following, affectionately nicknamed *Ford Nation*, along with strong support within Toronto city council. However, in 2013, Toronto city council's support for Mayor Rob Ford declined after highly publicised allegations of illicit drug use.

3.1 Canada’s House of Commons

3.1.1 Data Set

The Parliament of Canada is the federal bicameral legislature of Canada [12]. The legislature is divided into the upper house, called the Senate, and lower house, called the House of Commons [12]. Canadian MPs are the elected representatives who sit in the House of Commons and they hold more power in passing bills than its upper house counterpart [12]. The main parties are the Liberal Party, the Conservative Party, the New Democratic Party (NDP), the Bloc Québécois (BQ), and the Green Party (GP) [12]. Also, there can be MPs without any political affiliation, called Independent MPs [12].

The House of Commons’ website [16] provides voting records of the current and of past parliaments. On the website, there is a table for each bill that states whether the MPs voted ’yea’ or ’nay’ on that bill. Occasionally, MPs abstain from voting if they disagree with their party’s position on a particular issue, but abstaining votes are not formally documented [15]. From these tables we will extract three networks using three votes each. We chose votes with the intention of exhibiting parliaments with different party compositions.

First, we chose from the 40^{th} parliament 3^{rd} session which was a Conservative minority government with the Liberal Party was the official opposition. We used vote 3 which was introduced by a Conservative MP, vote 4 which was introduced by a Liberal MP, and vote 6 which was introduced by an NDP MP. All three of these bills were agreed upon. Second, from the 41^{st} parliament 2^{nd} session, we chose vote 1, vote 2, and vote 5. These bills were all introduced by a Conservative MP and were agreed upon. The 41^{st} parliament was a Conservative majority government; notably, with the NDP as the official opposition, which was the first time in the party's history. Third, from the 42^{nd} parliament 1^{st} session, we chose vote 12 which was introduced by an NDP MP, vote 14 which was introduced by a Conservative MP, and vote 17 which was introduced by a Liberal MP. All three of these bills were also agreed upon. The 42^{nd} parliament is the current Liberal majority government, with the Conservatives as the official opposition.

3.1.2 Methods

On the House of Commons' website [16], each table is a record of only those who voted on the bill. To simplify the data mining process, each graph includes only the MPs that were present to vote on the given bill. The graphs have n nodes representing the MPs and there is an edge connecting two nodes if they voted either both in favour or both against a bill. Each graph includes three bills, so the edges are weighted based on how many times two MPs voted together. For example, if two MPs voted the same way on two of the bills, then the edge connecting them is weighted 2.

In order to investigate what voting patterns reveal about the community structure of Canadian political parties, we will use the modularity score, the average degree, the clustering coefficients, centrality scores, and diameter. Each network will have four graphs with different partitioning results: modularity score, clustering coefficients, betweenness centrality, and closeness centrality. Each graph has a legend that details the colour of the nodes, the value of metric of those nodes, and the percentage of nodes with that value. Note that partitions based on clustering coefficients, betweenness centralities, and closeness centralities do not necessarily coincide with communities in the network. For all the graphs in this section we used the *Fruchterman Reingold* layout algorithm.

3.1.3 Results

In Figure 16 we have the first network, written N_1, constructed from three votes during the 40^{th} Parliament of Canada. Using Gephi, we obtained a modularity score of $Q_{N_1} = 0.492$ and the graph partitioned into two communities. The graph in Figure 16 shows N_1 partitioned based on modularity score. The blue community consists of the Conservative MPs, an Independent MP, and three BQ MPs, while the red community consists of Liberal MPs, NDP MPs, and the rest of the BQ MPs. The three BQ MPs that are in blue are the three nodes that connect the two communities. Note that there happened to have not been any GP MPs present at any of these three votes, thus there are no GP MP nodes in this network.

The graph in Figure 17 is partitioned based on the clustering coefficients. The light purple nodes are Conservative MPs. The bright green nodes are Liberal MPs, NDP MPs, and BQ MPs. The light blue nodes are Liberal MPs. The orange nodes are Conservative MPs, Liberal MPs, NDP MPs, and the Independent MP. The dark grey nodes are Liberal MPs and one BQ MP. The bright pink nodes are Liberal MPs and NDP MPs. Lastly, the teal nodes are the three BQ MPs that connect the two communities.

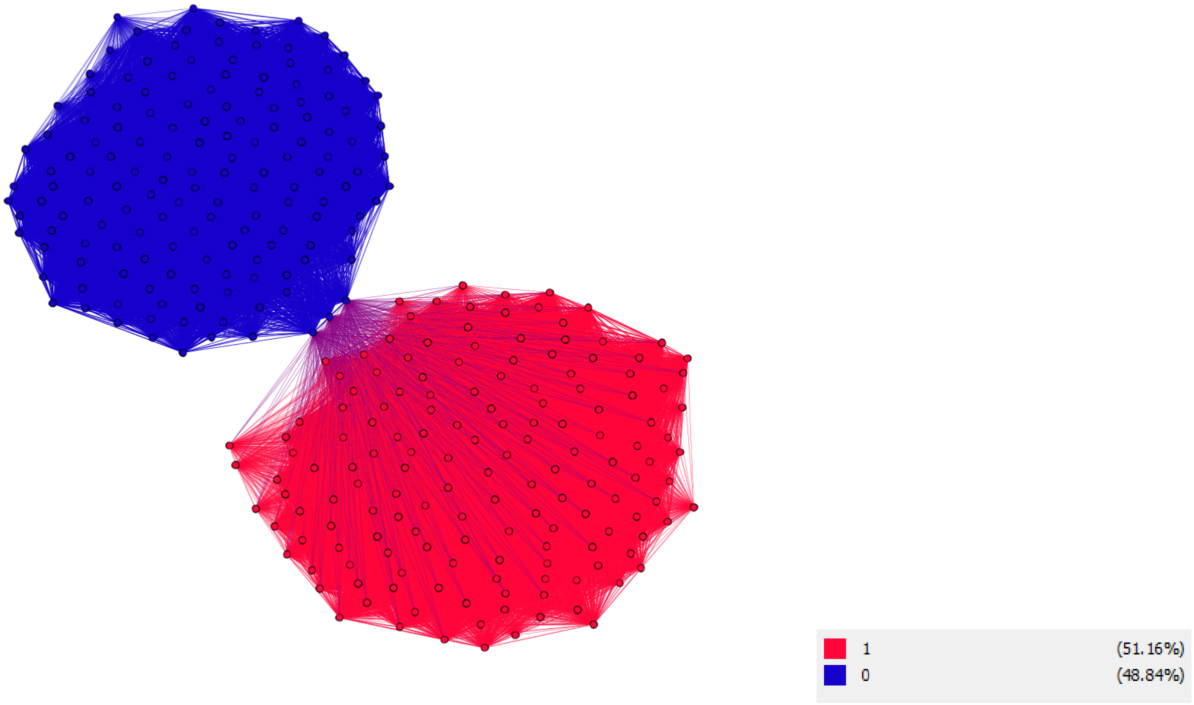


Figure 16: Modularity partitioning of N_1.

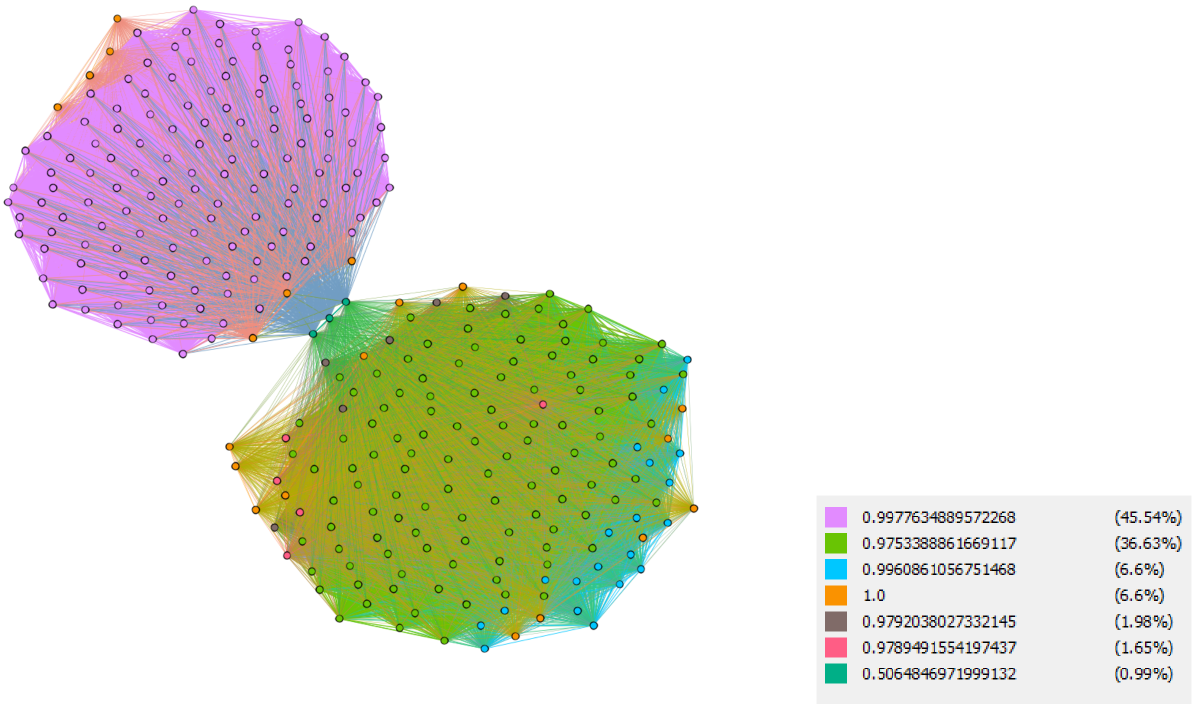


Figure 17: Clustering coefficient partitioning of N_1.

The graph in Figure 18 shows N_1 partitioned based on betweenness centralities. The light purple nodes are Conservative MPs. The bright green nodes are Liberal MPs, NDP MPs, and BQ MPs. The light blue nodes are Conservative MPs, Liberal MPs, NDP MPs, and the Independent MP. The orange nodes are Liberal MPs. The dark grey nodes are Liberal MPs and one BQ MP. The bright pink nodes are Liberal MPs and NDP MPs. Lastly, the teal nodes are the three BQ MPs that are connecting the two communities and have the highest betweenness centrality $C_b(v) = 7491.166$.

The graph in Figure 19 shows N_1 partitioned based on closeness centralities. The light purple nodes are Conservative MPs. The bright green nodes are Liberal MPs, NDP MPs, and BQ MPs. The light blue nodes are Liberal MPs. The dark grey nodes are Liberal MPs, NDP MPs, and one BQ MP. The orange nodes are Liberal MPs and NDP MPs. The bright pink nodes are Conservative MPs and the Independent MP. The teal nodes and the grey-pink nodes are Liberal MPs. The grey nodes are Conservative MPs. Lastly, the light grey nodes are the three BQ MPs that are connecting the two communities and have the highest closeness centrality $C_c(v) = 0.909$.

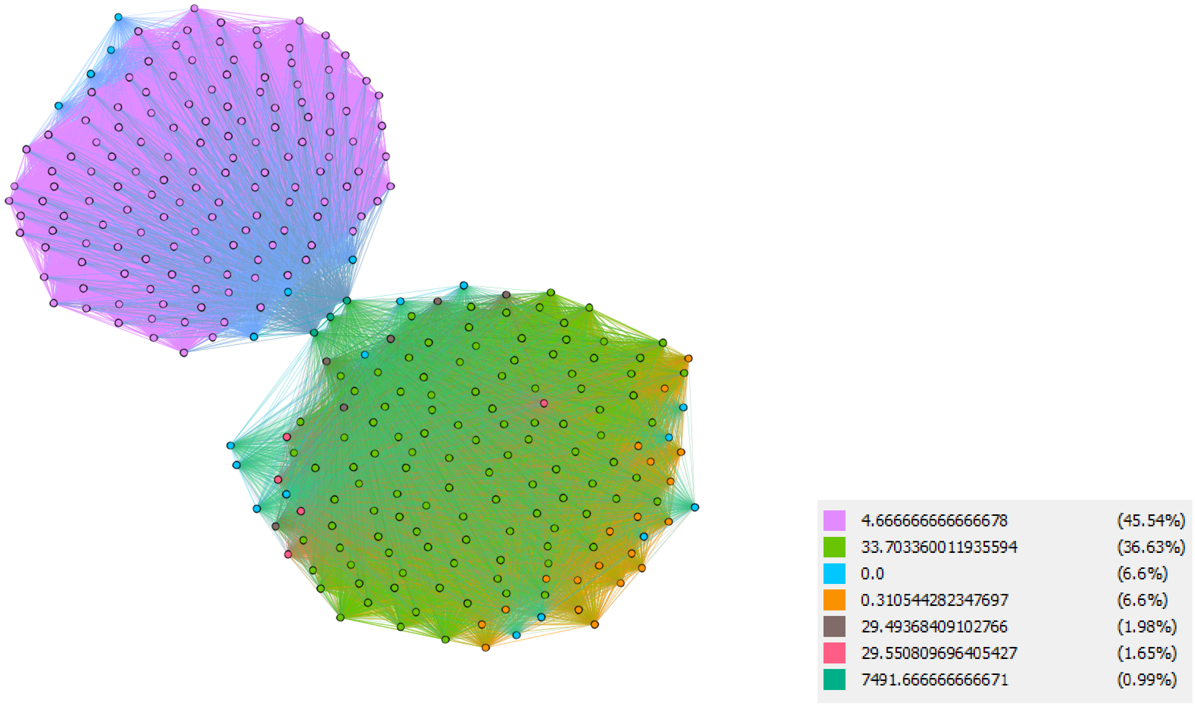


Figure 18: Betweenness centrality scores of N_1.

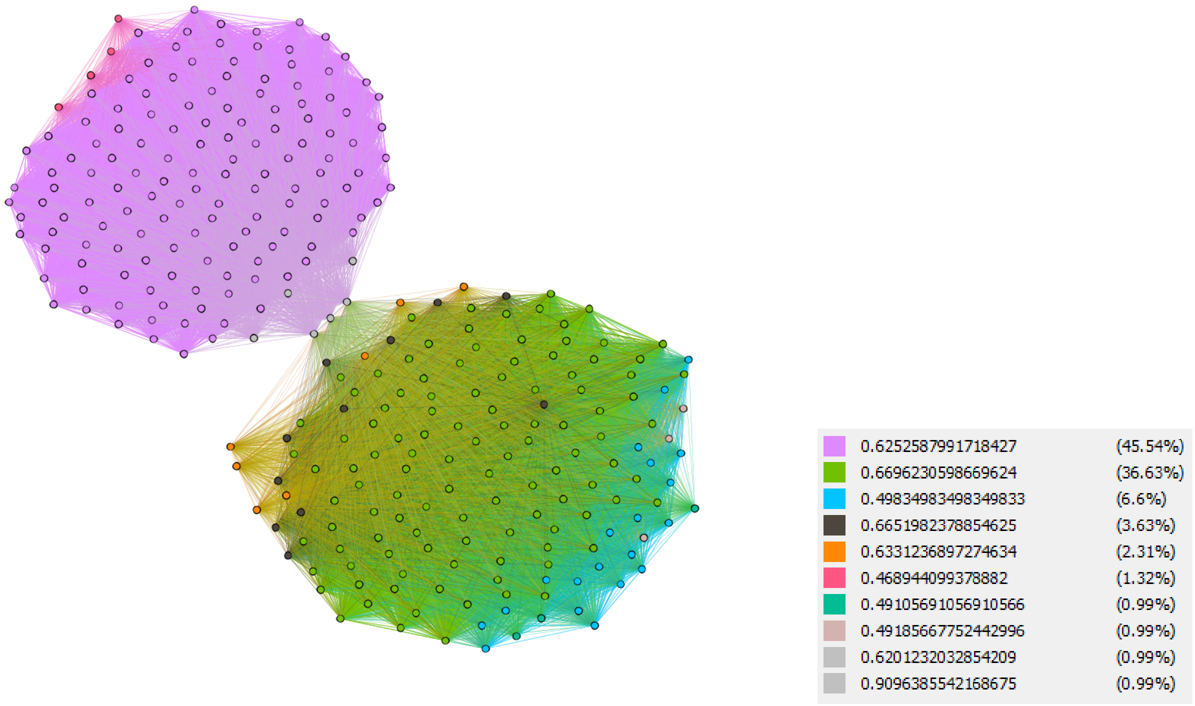


Figure 19: Closeness centrality scores of N_1.

The next network, written N_2, represents the three votes from the 41^{st} Parliament of Canada. The graph has a modularity score of $Q_{N_2} = 0.394$ and is partitioned into two communities. The graph in Figure 20 shows the Conservative MPs in blue and the Liberal MPs, the NDP MPs, the BQ MPs, the GP MP, and the Independent MPs in red. The nodes that are connecting the two communities are all Liberal MPs.

The graph in Figure 21 shows the partition based on clustering coefficients. The light purple nodes are Conservative MPs. The bright green nodes are Liberal MPs, NDP MPs, and the GP MP. The light blue nodes are Conservative MPs, Liberal MPs, NDP MPs, and BQ MPs. The orange nodes are Conservative MPs. The bright pink nodes are NDP MPs, BQ MPs, and Independent MPs. The grey-pink nodes are NDP MPs. The are two grey nodes that are Conservative MPs with clustering coefficient $C(v) = 0.987$ and one grey node that is a Conservative MP with $C(v) = 0.594$. The nodes that represent the Liberal MPs that connect the two communities are dark grey, teal, and light grey.

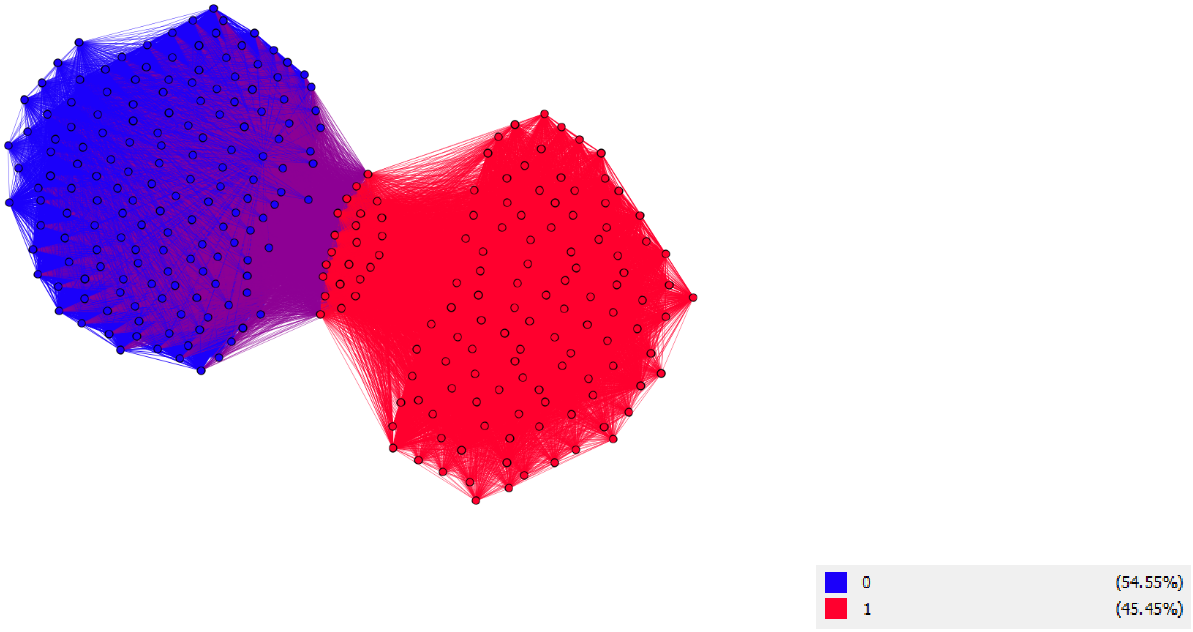


Figure 20: Modularity partitioning of N_2.

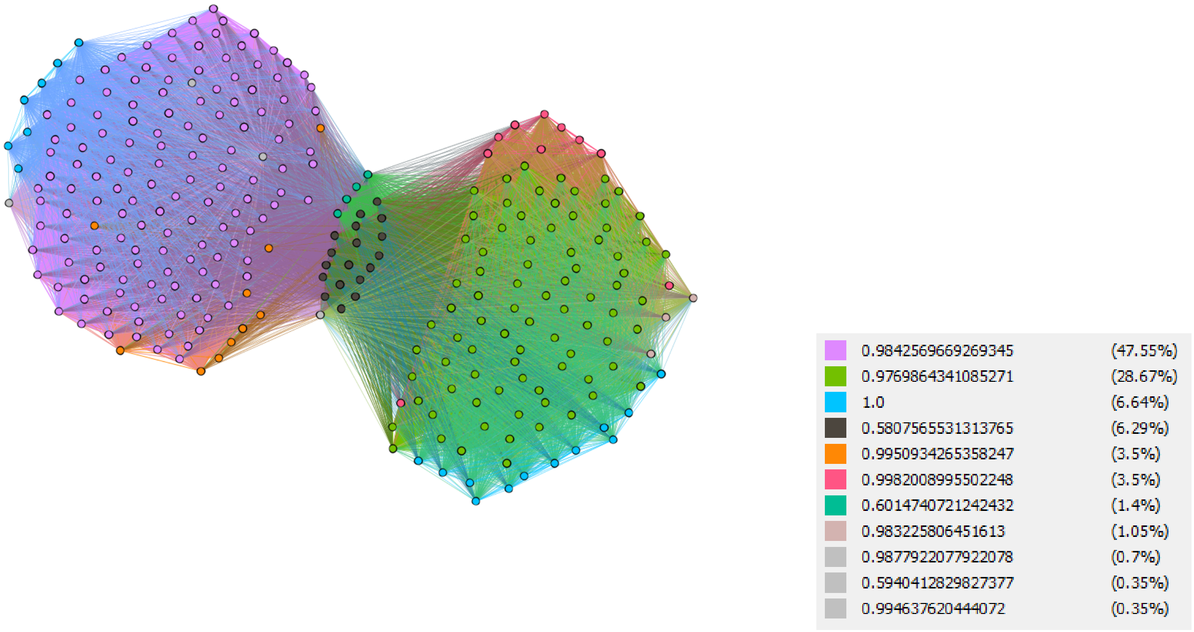


Figure 21: Clustering coefficient partitioning of N_2.

The graph in Figure 22 shows N_2 partitioned based on betweenness centrality. The nodes representing Conservative MPs are light purple, bright pink, light blue; as well as, the two grey nodes with betweenness centrality $C_b(v) = 5.948$ and one node with betweenness centrality 0.459. The Liberal MPs connecting the two communities are dark grey, teal, and light grey with the highest betweenness centrality scores $C_b(v) = 752.083$, $C_b(v) = 627.176$, and $C_b(v) = 679.915$, respectively. The rest of the Liberal MPs are bright green and light blue. The NDP MPs are bright green, light blue, orange and grey-pink. The BQ MPs are light blue and orange. The GP MP is bright green and the Independent MPs are orange.

The graph in Figure 23 shows N_2 partitioned based on closeness centrality. The nodes representing Conservative MPs are light purple, bright pink, light blue, teal; as well as, two grey nodes with closeness centrality $C_c(v) = 0.538$ and two grey nodes with closeness centrality $C_c(v) = 0.723$. The Liberal MPs connecting the two communities are light blue, grey-pink, and grey with the highest closeness centrality scores $C_c(v) = 0.972$, $C_c(v) = 0.925$, and $C_c(v) = 0.940$, respectively. The rest of the Liberal MPs are bright green and dark grey. The NDP MPs are dark grey, bright green, orange and grey. The GP MP is light green and the Independent MPs are orange.

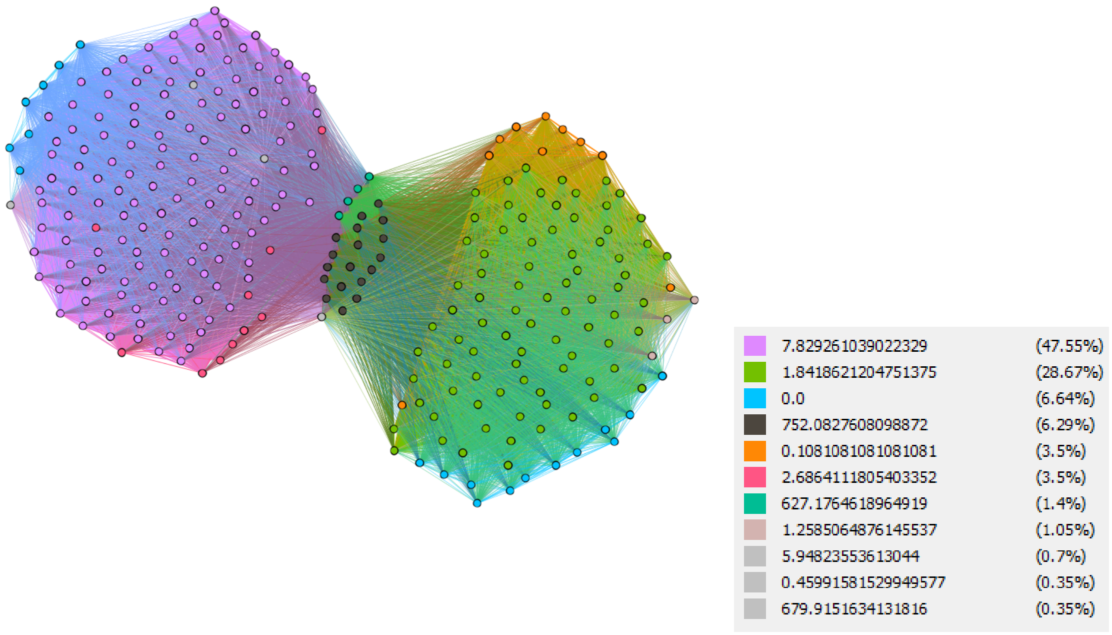


Figure 22: Betweenness centrality scores of N_2.

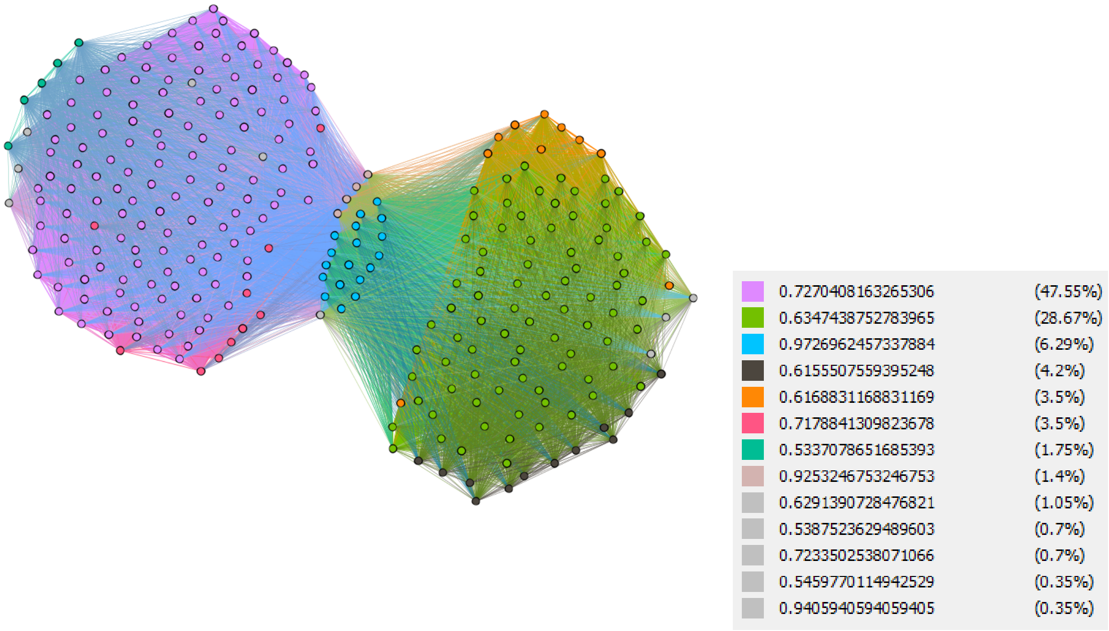


Figure 23: Closeness centrality scores of N_2.

The last network in of this section, written N_3, represents the three votes from the 42^{nd} Parliament of Canada. It has a modularity score of $Q_{N_3} = 0.198$ and is partitioned into two communities. The the graph in Figure 24 shows the Conservative MPs and some Liberal MPs in blue, and the rest of the Liberal MPs, the NDP MPs, the BQ MPs, and the GP MP are in red. The cluster of red nodes in the centre of the graph are all Liberal MPs.

The graph in Figure 25 shows the partition based on the clustering coefficients. The light purple nodes are Liberal MPs. The bright green nodes are Liberal MPs, NDP MPs, BQ MPs, and the GP MP. The light blue nodes are Conservative MPs. The dark grey nodes are Conservative MPs, Liberal MPs, and BQ MPs. The bright pink and orange nodes are Conservative MPs. The teal nodes are NDP MPs, one Liberal MP, and one BQ MP. The grey-pink nodes are Conservative MPs. Out of the nodes representing the Liberal MPs connecting the two communities, there are two grey nodes with clustering $C(v) = 0.823$ and four grey nodes with clustering coefficient $C(v) = 0.998$.

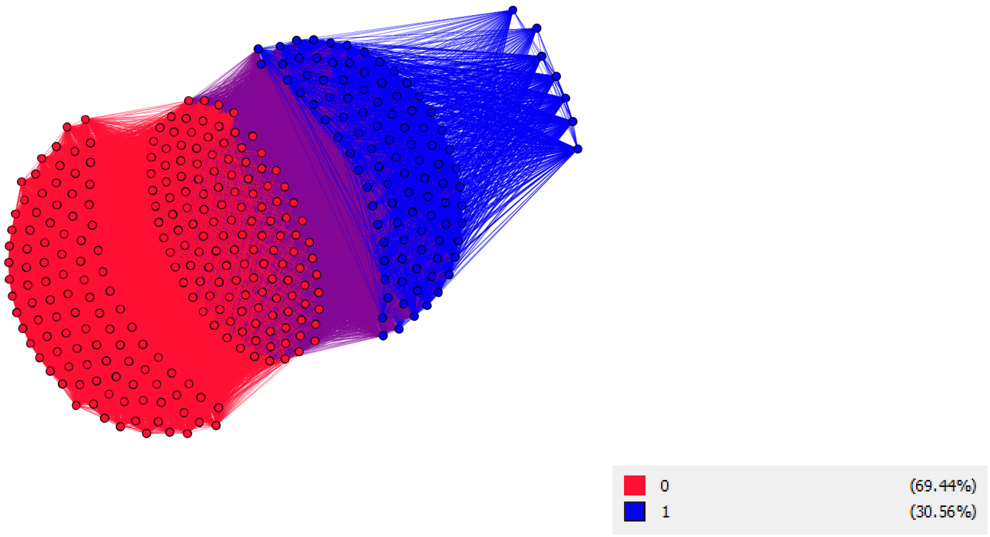


Figure 24: Modularity partitions of N_3.

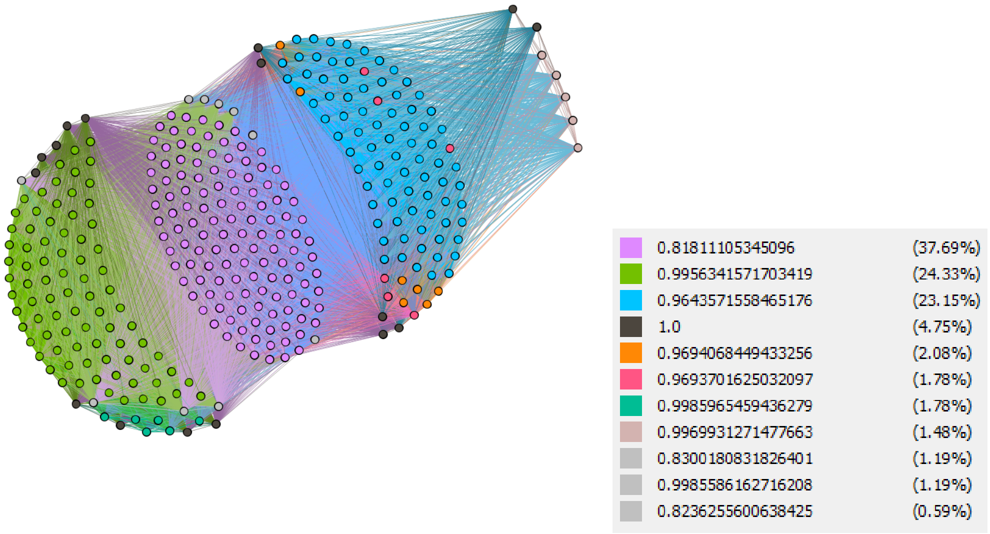


Figure 25: Clustering coefficient partitioning of N_3.

The graph in Figure 26 shows N_3 partitioned based on betweenness centrality. The Conservative MPs are light blue, orange, dark grey, and grey-pink. The Liberal MPs in the blue community are dark grey. The Liberal MPs that are connecting the two communities are light purple and grey, and have the highest betweenness centrality scores of $C_b(v) = 79.154$ and $C_b(v) = 74.128$, respectively. The Liberal MPs in the red community are dark grey and light green. The NDP MPs are light green, grey, and pink. The BQ MPs are dark grey, pink, and grey. Lastly, the GP MP is light green.

The graph in Figure 27 shows N_3 partitioned based on closeness centrality. The Conservative MPs are light blue and grey. The Liberal MPs in the blue community are grey. The Liberal MPs that are connecting the two communities are light purple and grey, and have the highest closeness centrality scores of $C_c(v) = 0.979$ and $C_c(v) = 0.943$, respectively. The rest of the Liberal MPs and the BQ MPs are bright green, pink, orange and grey. The NDP MPs are bright green, pink and grey. Lastly, the GP MP is bright green.

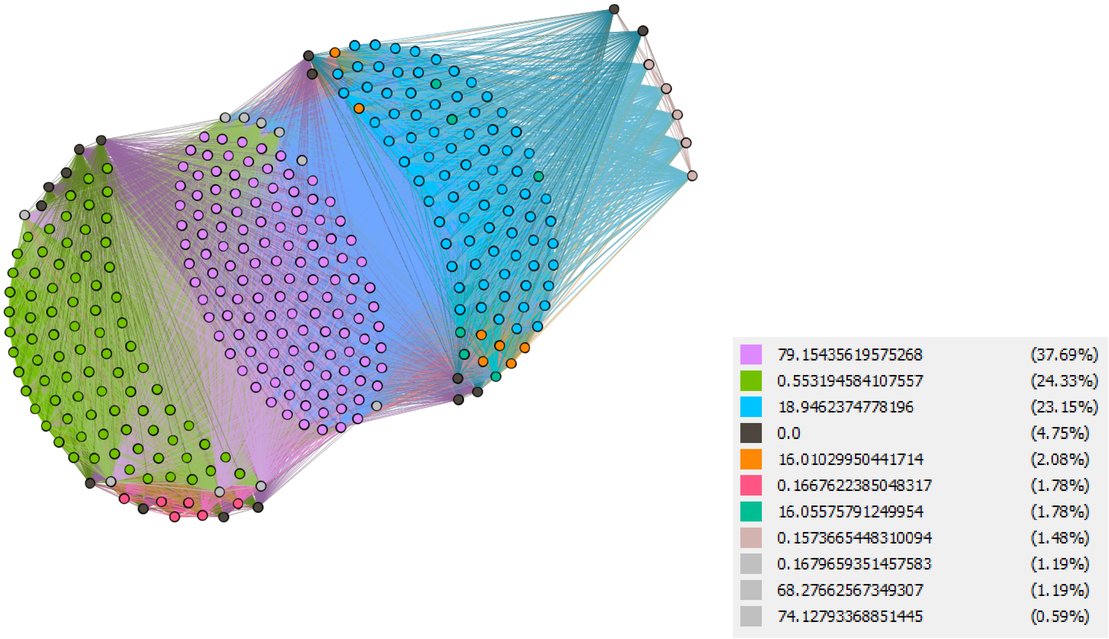


Figure 26: Betweenness centrality scores of N_3.

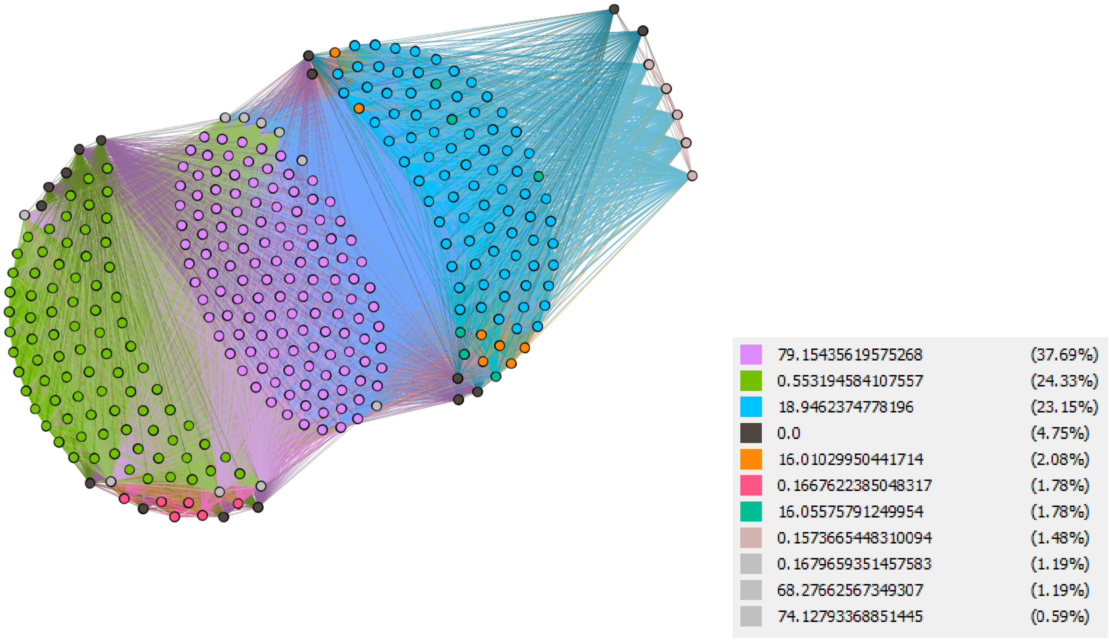


Figure 27: Closeness centrality scores of N_3.

The following table summarises the average degree, diameter, average path length, and average clustering coefficient for each network.

Network	$\text{deg}_{\text{av}}(\text{G})$	$\text{diam}(G)$	$L(G)$	$C(G)$
N_1	151.498	4	1.594	0.984
N_2	165.210	3	1.441	0.952
N_3	267.745	3	1.216	0.918

3.1.4 Discussion

By considering our visualizations of N_1, N_2, and N_3, we see that the community structure that arises across the three networks is a partitioning between two main communities. Namely, the governing party and those that vote with it, and the official opposition and those that vote with it. Previous research suggests that there are two *dimensions* that explain voting decisions. The first dimension is voting decisions based on right-left ideology and the second is decisions based on more specific agendas, such as ethnic, linguistic or regional interests [15]. The Conservative Party lies right of centre on the ideology spectrum and the NDP lies left of centre; whereas, the Liberal Party falls somewhere in the middle. They are known as a 'catch-all' party and fluctuate between right and left of centre depending on the issue at hand [12].

Figure 16 shows a clear distinction, in network N_1, between the MPs voting with the governing Conservative party and the MPs voting with the official opposition, the Liberal party. Here, it is interesting that the three nodes connecting the two communities represent BQ MPs. As all three networks show, typically the BQ votes with the NDP. This can maybe be explained by regional interests. The BQ has a history of opposing the Liberal Party when it comes to issues regarding Québéc [12]. This cluster of three nodes had both the highest betweenness centrality score and closeness centrality score. Thus, these three nodes are highly connected to the rest of the network; however, they are not apart of a dense cluster. They are significant to the network because since the 40^{th} parliament was a Conservative minority government, if the Liberals wanted to pass a bill it would be in their best interest to have all the BQ MPs vote with them.

Figure 20 shows that N_2 is partitioned into two communities: the Conservative government and its NDP official opposition. All of the BQ MPs; as well as, some of the Liberals, voted with the NDP. However, here, most of the Liberal MPs are in the cluster connecting the two communities. This network coincides with the ideologies of the parties. The NDP is on side and the Conservative party is on the other, while the Liberal party are in the centre. Like in the previous analysis, the nodes clustered in the middle have the highest betweenness centrality and the highest clustering centrality, but have a low clustering coefficient of $C(v) = 0.581$. These nodes are important to the network because they are highly connected but not apart of a dense cluster. Therefore, since the 41^{st} parliament was a Conservative majority government, their support would be essential for any of the non-governing parties to pass a bill.

In Figure 24, we still see two communities in N_3. However, in the graph that is partitioned by clustering coefficient, we can see a distinction between the primarily light purple cluster of Liberal MPs and the primarily bright green cluster, made up of Liberal, NDP and BQ MPs. This network appears to illustrate a less polarized government and opposition than N_1 and N_2. This is not unusual since N_3 represents our current Liberal majority government and it demonstrates the 'catch-all' characteristic of the Liberal party. Most of the Liberals are in the centre of the ideological spectrum, while some Liberals side with the Conservatives and some side with the NDP and BQ.

Notice that the diameters of N_2 and N_3 are both $\text{diam}(N_2) = \text{diam}(N_3) = 3$; whereas, the diameter of N_1 is $\text{diam}(N_1) = 4$. This suggests that N_1 represents the most polarized of the three parliaments. Since N_1 was a Conservative minority the more moderate and left-leaning MPs needed to ally in order to pass a bill or stop a bill they disagreed with. The analysis of these three networks implies that the Liberal party votes more with the opposition the more power the Conservatives have.

These three examples of networks obtained from the voting patterns of Canadian MPs show distinct communities that form along informal alliances. For the most part, the community structure corresponds to where the parties land on the ideological spectrum.

This was shown with as little as three votes per parliamentary session; however, future work could include all votes of a parliamentary session, which would hopefully provide more details about the structure of the network and the networks important members.

3.2 Ford Nation

3.2.1 Data Set

The Toronto city councillors are not part of a political party like Canadian politicians on the provincial or federal level are; however, they are elected to represent the interests of distinct ward boundaries. Rob Ford served as a city councillor representing Ward 2 of Etobicoke North, then as mayor of Toronto from 2010-2014 [6]. He received a lot of attention due to allegations of illegal drug use after a video surfaced in March of 2012 [10]. The support of Ford Nation stayed relatively strong throughout his time as mayor, but Ford's support within city council started to decline in 2013 to the point where they removed his key responsibilities as mayor. Nevertheless, Mayor Ford did initially have consistent support amongst his fellow councillors.

In this section, we consider the voting patterns of the Toronto city councillors from the beginning of Rob Ford's term as mayor. We used data from a table found on-line [9] that is comprised of twenty votes on key issues from 2011. The rows are each of the city councillors and the columns are each of the twenty votes. If a councillor voted the same way was Rob Ford on a vote, then their record for that vote would be a green 'yes' and if they did not vote the same way, then their record for that vote would be a red 'no'.

3.2.2 Methods

We created two graphs using the city councillor voting data. The first graph has 45 nodes, one for each city council member including Mayor Ford. There is an edge between two councillors if they voted the same way as Rob Ford. This means that there is an edge between Rob Ford and another councillor if they voted the same way and there is an edge between two councillors if they both voted the same way as Rob Ford. Note that here we do not consider if each councillor voted in favour or against the issue, just if they voted the same way as Mayor Ford. The edges are weighted for how many times two councillors vote the same way as Ford. Since the data considers twenty votes, the edge weights can be up to weight 20.

Then using that same twenty votes, we also generated a dynamic graph. The same nodes and edges are defined the same way. However, instead of it being weighted for how many times two councillors voted with Ford, each vote has a timestamp. For example, the vote on 'reduce councillor expense budget' will be time 1 and the vote on 'eliminate vehicle registration tax' will be time 2. This creates a dynamic graph where we can observe the city councillors voting patterns across twenty votes. To measure the support of Mayor Ford, we used the clustering coefficients and the modularity score. For all the graphs in this section we used the Fruchterman Reingold layout algorithm.

3.2.3 Results

We used Gephi to visualize the graph and obtained a modularity score of $Q = 0.048$. The network was partitioned into two communities, as seen in Figure 28. Note that the giant node represents Rob Ford. The community that is shown in pink consists of the councillors that voted most often with Rob Ford and has clustering coefficient $C(v) = 1.0$. The community in black are those that voted less often with him and have clustering coefficient $C(v) = 0.96$. The average clustering coefficient of the graph is $C(G) = 0.98$.

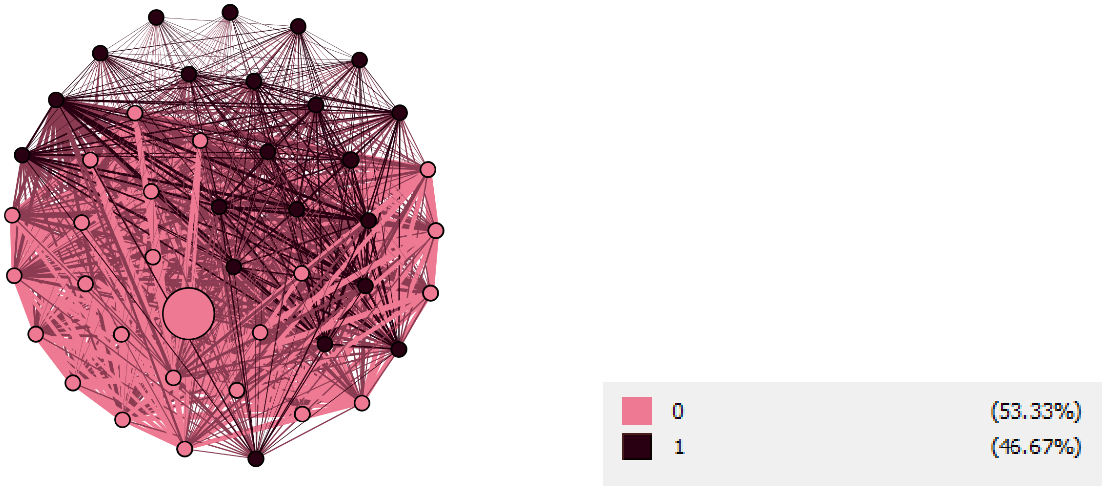


Figure 28: Partitioning based on modularity score.

In Figure 29, the graph is partitioned based on clustering coefficients. The large mauve cluster has clustering coefficient of $C(v) = 0.9820$ and it contains the nodes from the pink community surrounding Rob Ford in Figure 7.

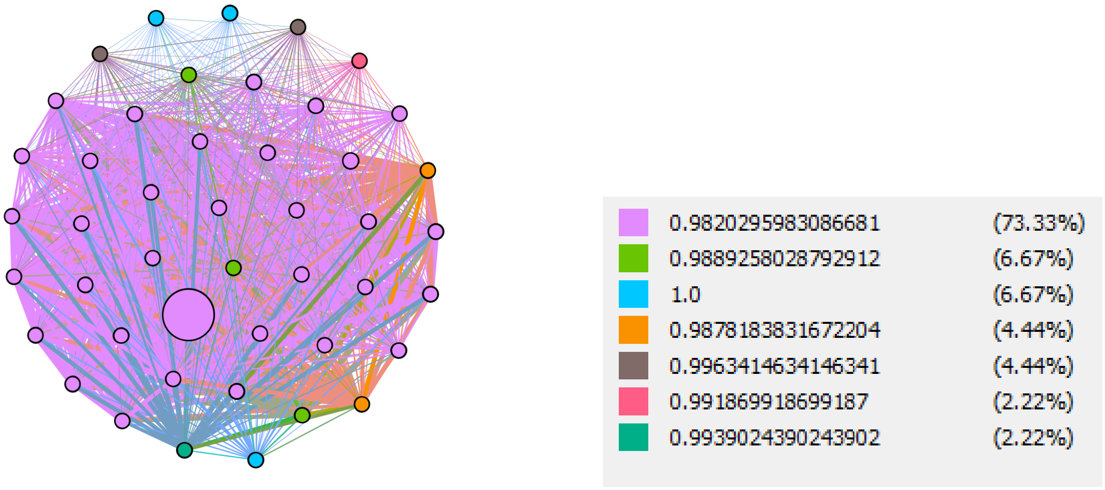


Figure 29: Partitioning based on clustering coefficients.

The graphs in Figure 30 show the dynamic version of the Rob Ford graph where each graph is a timestamp representing a separate vote. The nodes in pink are the same community as obtained from the partitioning using the modularity score in Figure 28. The graphs should be interpreted left to right and top to bottom.

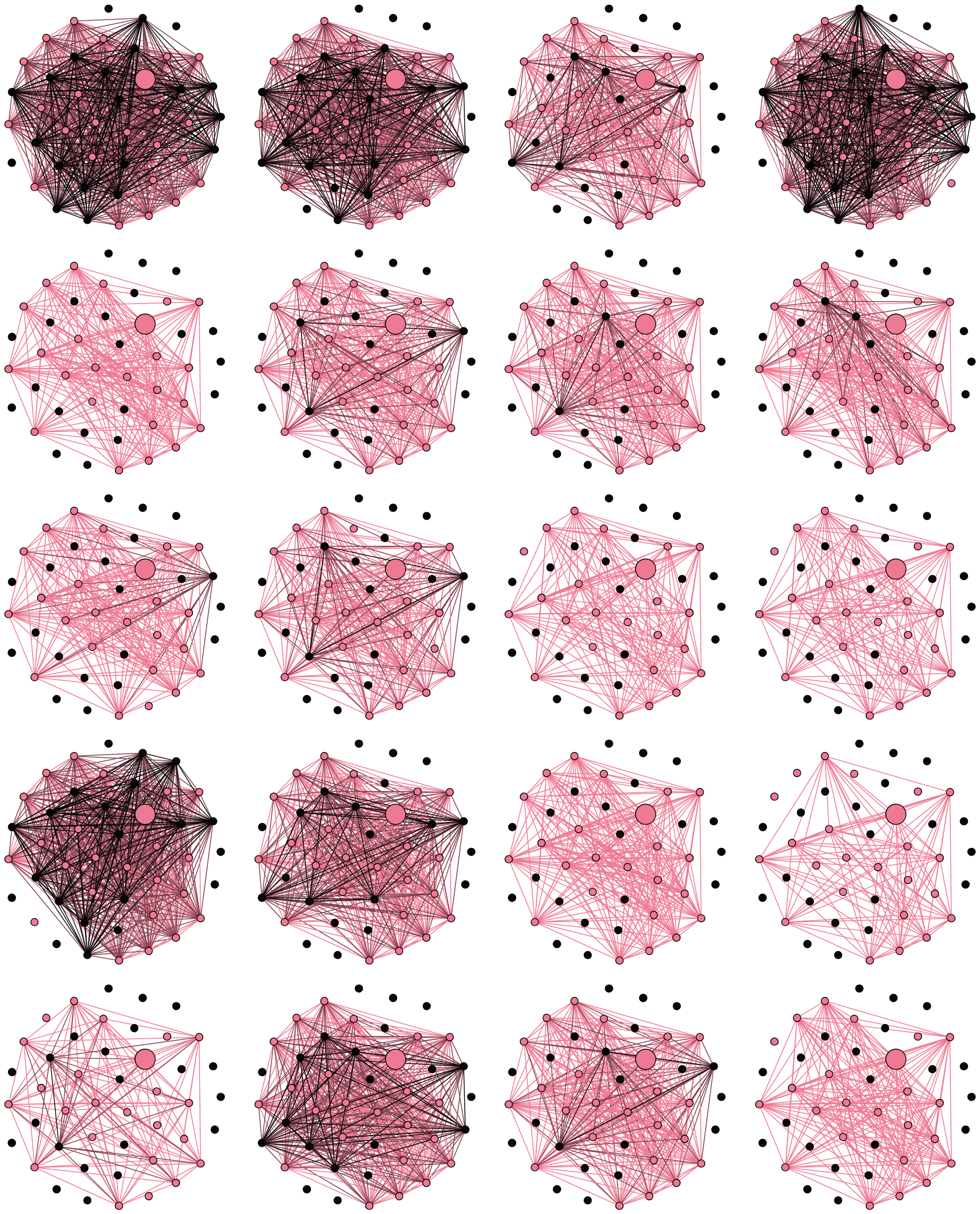


Figure 30: Dynamic graph of city councillors voting with Rob Ford over twenty votes.

3.2.4 Discussion

When looking at the data in table format, it is unclear how to interpret the data. However, in Gephi, the visualization of the network illustrates definitive communities. In Figure 28, we see community which surrounds the Rob Ford node in pink and the other community in black. The high clustering coefficient for the community surrounding Rob Ford shows that he had strong support from city council back in 2011, long before the infamous video surfaced. The clustering coefficient for the community coloured in black is also high, indicating that those councillors voted consistently less with Rob Ford across the twenty votes.

The dynamic graph, in Figure 30, emphasizes the presence of the community of councillors that consistently supported Mayor Ford early in his term. Although there are some votes where the vast majority of the councillors voted with Rob Ford, the dynamic graph as a whole shows that there is consistency when it comes to those that voted with Ford and against him. Rob Ford seemed to have unwavering support from Ford Nation and the dynamic graph shows the reflection of that support within city council in 2011.

While Rob Ford always had a reputation of being brash, after the allegations of illicit drug use his support among the city councillors dramatically declined. City councillors are not allowed to formally vote the mayor out of office, but at the end of October 2013 the councillors had an informal and unsuccessful vote to encourage Mayor Ford to step aside [10]. Consequently, in November of 2013, councillors voted to take away many of Ford's responsibilities, leaving his powers as mayor mostly symbolic [11]. Despite his colleagues having lost confidence in his governing abilities, Ford Nation remained resilient. This was made evident during Toronto's 2014 mayoral election and 2014 municipal election. Rob Ford ran for re-election, but dropped out of the running for mayor after being diagnosed with cancer [6]. He endorsed his brother, Doug Ford, who took over the campaign and came in second with 33.7 percent of the vote, after John Tory with 40.3 percent and ahead of Olivia Chow with 23.2 percent [6]. Subsequently, Rob Ford ran for his old city council seat representing Ward 2 and won [6].

Future work would be to explore the discrepancy between public support and city council support for Rob Ford after the drug allegations scandal. In additon, Figure 23, shows more nuanced structure of the Toronto city council during Rob Fords term as mayor. Alliances between councillors that have similar values should also be explored because they could reveal hidden political affiliations.

4 International Political Networks

In the previous chapter we discussed networks related to Canadian politics, specifically to networks arising from voting patterns. Here we shall study networks linked to topics concerning international politics. Interactions between nations differ from country to country and a nation's ability to play a substantial role on the international stage can be affected by a number of factors. For instance, it can be affected by geography, by wealth, or by historical ties. Therefore, it is challenging to generalize relationships between countries. One of the indicators of a good relationship between countries is trade; namely, arms trade [2]. High quantities of bilateral arms trade is seen a symbol of trust between two nations [2].

There are many important world actors including the United States (U.S.), Russia, China, and Germany. The U.S. government is an especially important world actor and its leader receives a lot of international attention. Thus, it was controversial when real-estate mogul and reality star, Donald Trump was elected president in 2016. His unfiltered use of Twitter has created a sensitive relationship between the current U.S. government and its citizens; as well as, changed the dynamics of U.S. foreign relations. In this chapter, we will explore two different examples of using network science to analyze topics related to international politics.

First, we will consider trade networks using records of all international arms trade and of oil exports from Africa. Inspired by the article [2], which used records of all arms trade from 2006 to 2015, we recreated their graph with the intention of expanding on their analysis. We compare this graph to that of oil exports from Africa and discuss what the community structures indicate about alliances between countries. Moreover, we will be using a planar graph of the map of Africa to investigate the impact of geography on the alliances.

Second, we will consider linguistic networks extracted from President Trump's personal Twitter account: @realDonaldTrump. Here we will explore the evolution of Donald Trump's use of Twitter from presidential candidate to president elect to president of the U.S, focusing on the topics he communicates to his constituents and to the rest of the world.

4.1 Arms for Oil

4.1.1 Data Set

In this section we will be studying arms trade and oil trade, specifically surrounding foreign relations of African countries. We used records of international arms trade from the Stockholm International Peace Research Institute (SIPRI) website [27]. The website allows for the data to be filtered based on supplier, recipient, year, and type of weapons. They use the conventional measurement of *trend-indicator value* (TIV) [27]. The TIV is used to quantify the value of the trade in terms of resources rather than directly in terms of cost [27]. We used records of all weapons trade from 2006 to 2016, including weapons such as aircrafts, missiles, and naval weapons [27]. The graph we created using this SIPRI data was inspired by the article [2], where they used records from 2006 to 2015; however, here we were also able to add records from the year 2016.

Then we used records of international oil trade from [29] provided by the International

Trade Centre. The website provides a table that can be filtered based on imports, exports, country, year, currency, and product. We searched the records labelled as *Petroleum oils and oils obtained from bituminous minerals, crude* exported from African countries in 2016 in U.S. dollars [29]. Finally, we used an image of the continent of Africa to create a planar graph [1].

4.1.2 Methods

The first network, written N_{a1}, is an undirected graph where the nodes represent countries and there is an edge between two nodes if they exchanged arms valuing one hundred million TIV or more from 2006 to 2016. Each normalized edge weight based on the TIV amount traded in millions. Not all countries participated in arms trade during these years, thus there are 95 nodes in N_{a1}. To study this network we used the modularity score, clustering coefficient, and degree. We used an undirected graph since the data includes all weapons trade. Many countries import and export different to each other; therefore, the edge weights represent the cumulative total of trade in either direction between two countries. The size of the nodes represent the degree and should be interpreted as the nodes with higher degree represent the countries that export more arms than import. We visualized N_{a1} in Gephi using the *Yifan Hu* [18] layout algorithm with Label Adjust.

We also discuss a subgraph of N_{a1}, written N_{a2}. In the article [2], they use a metric called *PageRank* [23] to determine the top fifteen countries in the graph with the most international influence. The top fifteen are the U.S., Russia, Germany, France, China, Ukraine, the Netherlands, Italy, the United Kingdom (U.K.), Spain, Sweden, Israel, Turkey, India, and Pakistan [2]. Thus, network N_{a2} includes the nodes representing African countries found in N_{a1} and the top fifteen influential countries in terms of arms trade determined by [2].

The next graph, written N_p, represents the network of oil exports from African countries to countries outside of Africa. Again, not all countries import oil from Africa, thus there are 89 number of nodes. Each node represents a country and there is an edge between two nodes if there is oil traded between the two countries. The edges have normalized weights based on the U.S. dollar amount of the trade in thousands. The network N_p is a directed graph, with the African country as the tail of the arrow and the head of the arrow going toward the country it exports oil to. Again, the size of the nodes represent the degree. To visualize N_p we used the Yifan Hu layout algorithm with Label Adjust, then used the modularity score, clustering coefficient, and average degree.

Finally, we have a planar graph of Africa, where each node is a country in Africa and there is an edge between two nodes if they are neighbouring countries. The graph is unweighted and undirected. We include this graph twice. Once with the colouring from the partitioning of N_{a1}, written N_{map1}, and second time with the colouring from the partitioning of N_p, written N_{map2}. The planar graph was laid out out manually to show best show the communities in terms of geography.

4.1.3 Results

In Figure 31, we have N_{a1} partitioned into three groups with modularity score $Q_{N_{a1}} = 0.384$, average clustering coefficient $C(G) = 0.537$, and average degree $\deg_{\text{av}}(G) = 6.723$. The three groupings are shown by different colours. The green community which includes the countries that trade arms mostly with Russia and China. The blue community which includes the countries that trade mostly with the U.S. and France. As well as, the orange community which includes the countries that trade mostly with the Netherlands and Germany.

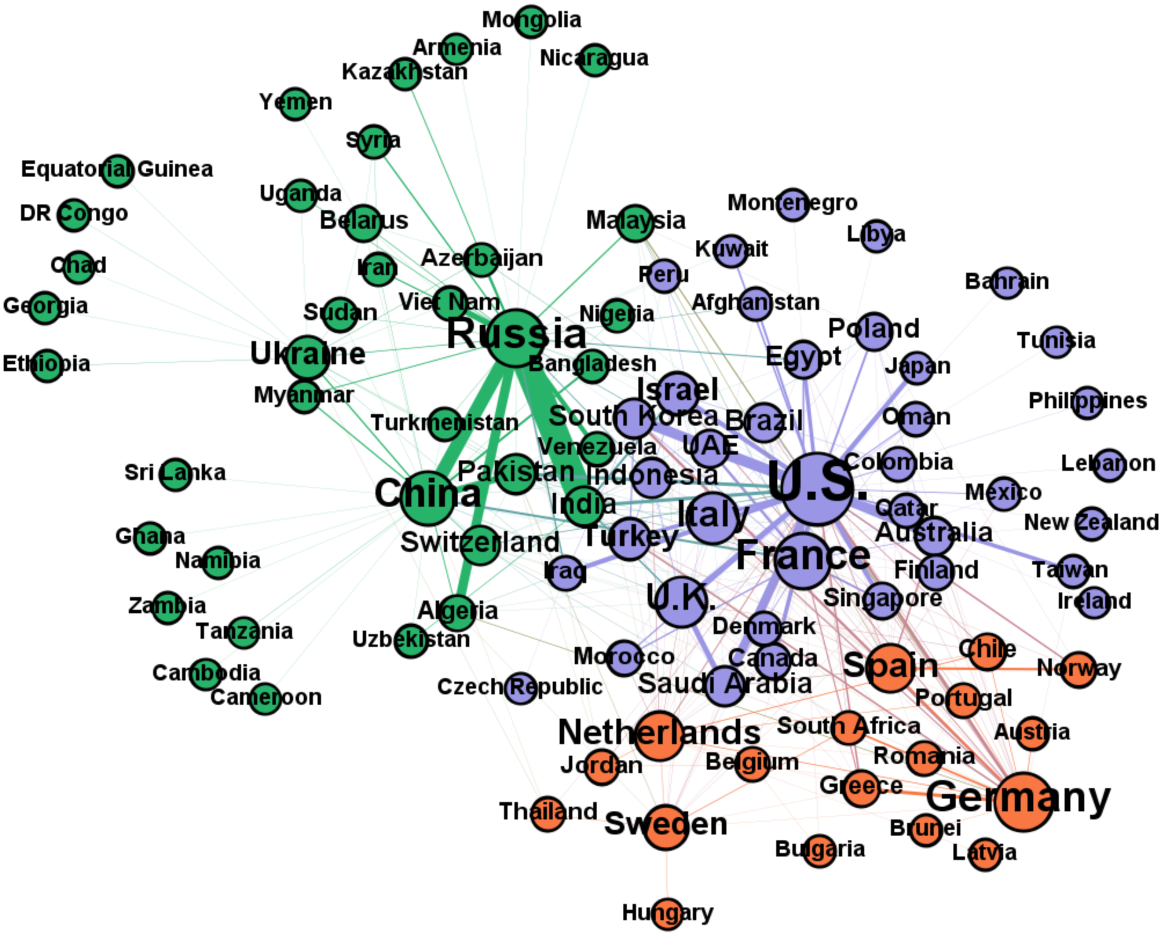


Figure 31: Network N_{a1} of arms trade with a value of one hundred million TIV or more.

In the following, Figure 32, we have N_{a2} which is a subgraph of N_{a1}. It has modularity score $Q_{N_{a2}} = 0.365$, average clustering coefficient $C(G) = 0.508$, and average degree

$\mathrm{deg}_{\mathrm{av}} = 6.000$.

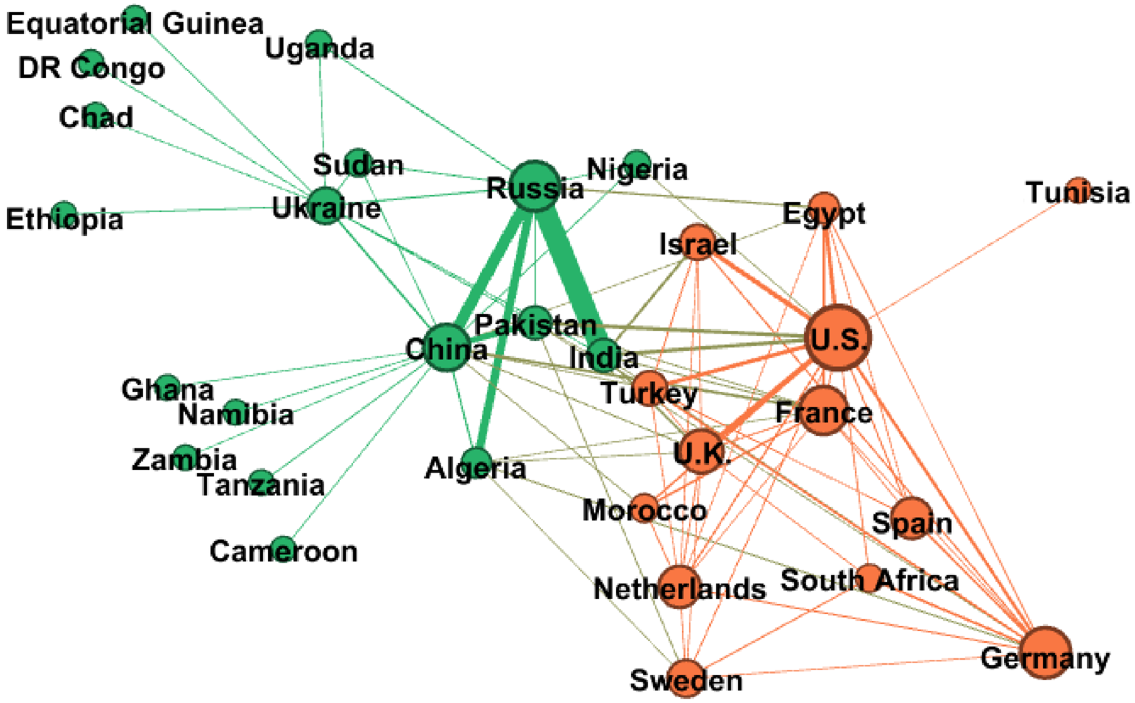


Figure 32: Network N_{a2} which is a subgraph of N_{a1} with only the African countries and top fifteen influential countries.

In Figure 33, we have N_p partitioned into five groups with modularity score $Q_{N_p} = 0.377$, average clustering coefficient $C(G) = 0.052$, and average degree $\mathrm{deg}_{\mathrm{av}}(G) = 2.663$. The purple nodes are the countries that imports oil mostly from Nigeria and South Africa. The blue nodes are the countries that import mostly from Angola. The orange nodes import mostly from Libya. The green nodes import from mostly Algeria and Equatorial Guinea. The three teal nodes are Namibia, Botswana and Russia.

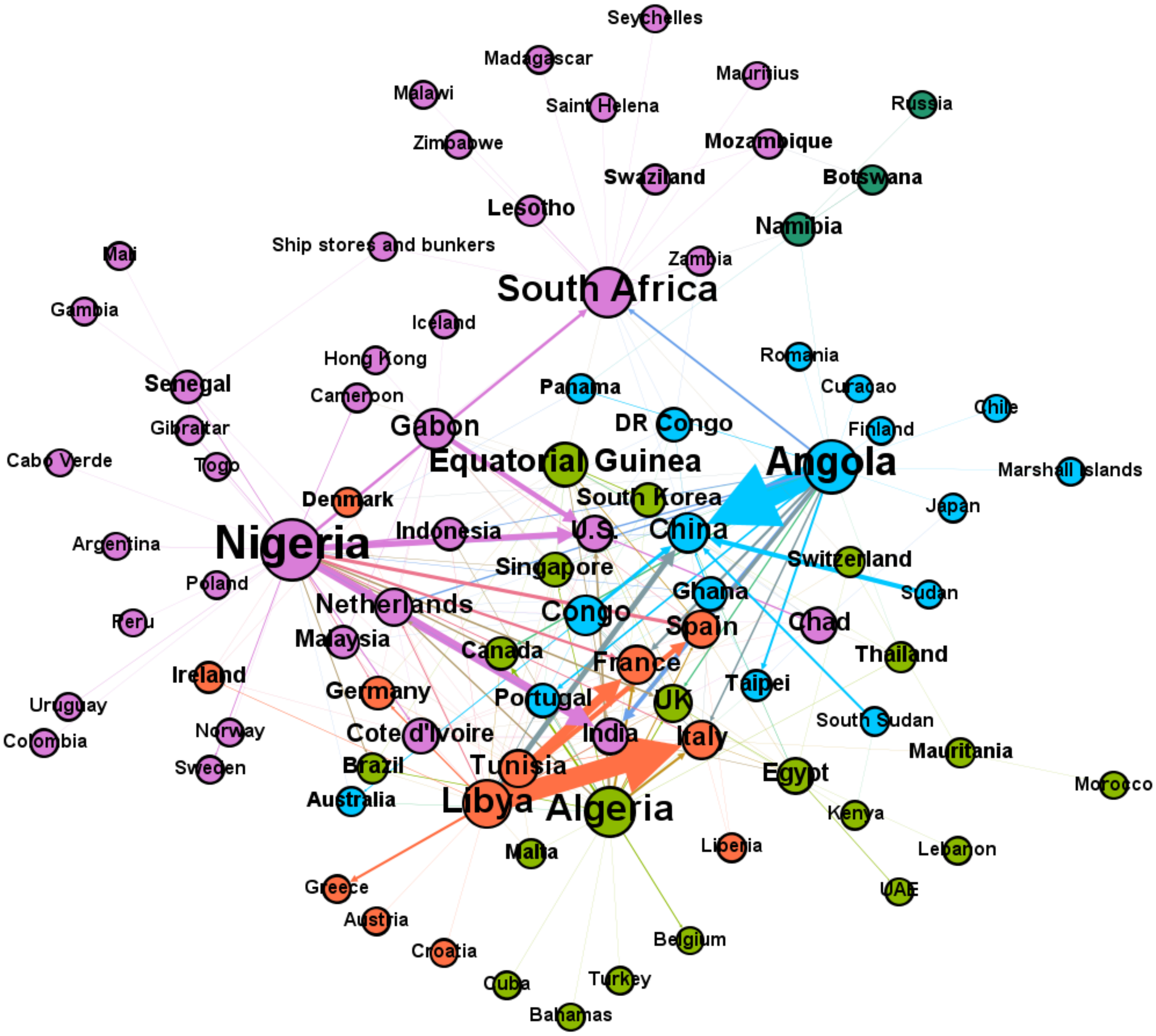


Figure 33: Network N_p of oil exported from African countries.

Next, Figure 34 shows N_{map1}, the planar graph of the map of Africa coloured using the partitioning from the modularity sore in N_{a1} from Figure 25. Recall that the green community contains Russia and China, the blue community contain the U.S. and France, and the orange community contains the Netherlands and France.

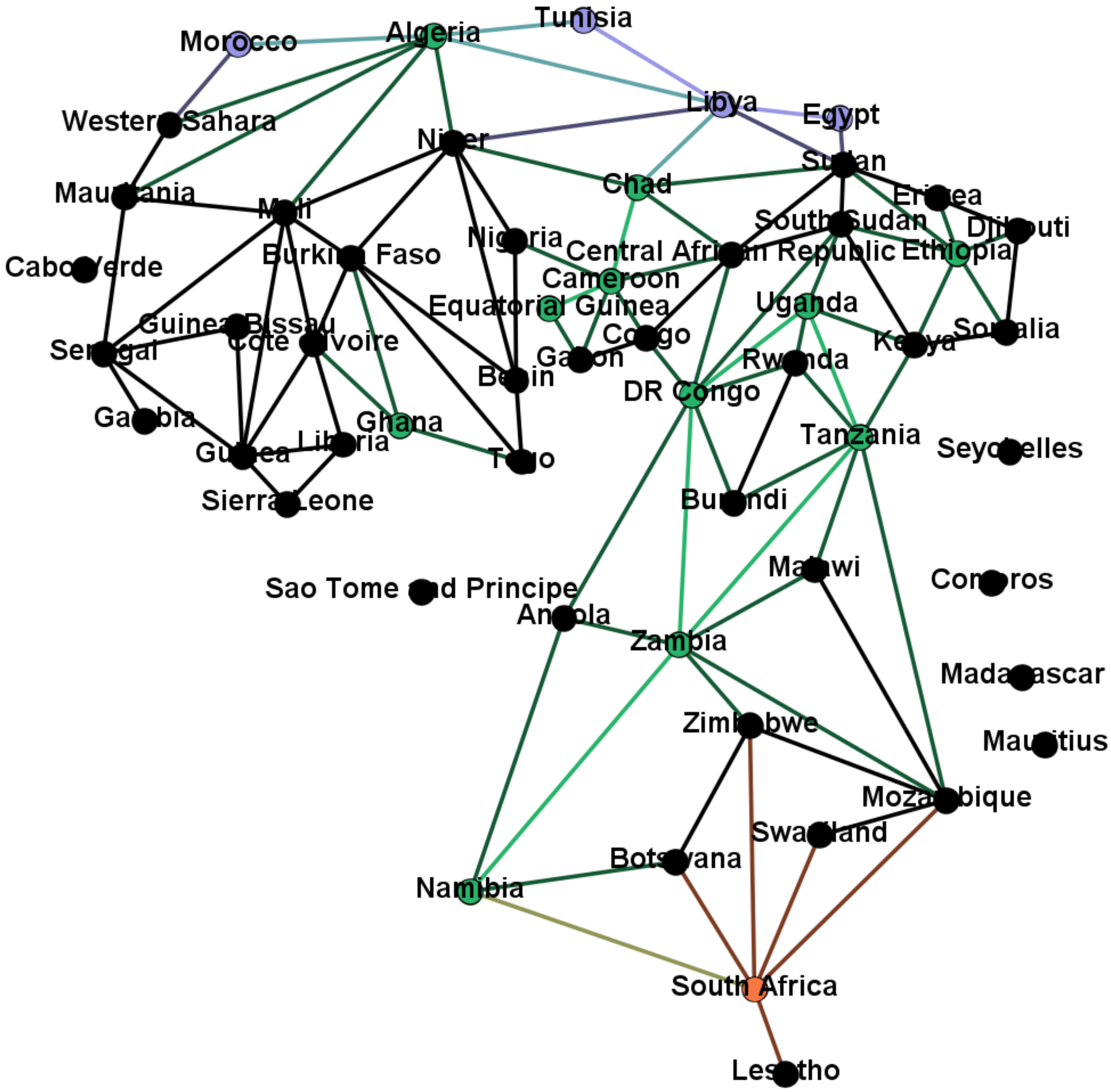


Figure 34: Planar graph, N_{map1}, with the node colouring of N_{a1}

Lastly, Figure 35 shows N_{map2}, the planar graph of the map of Africa coloured using the partitioning from the modularity sore in N_p from Figure 33. Note that in N_p Russia was in the teal community, China was in the blue community, Germany and France were in the orange community, and the U.S. and the Netherlands were in the purple community.

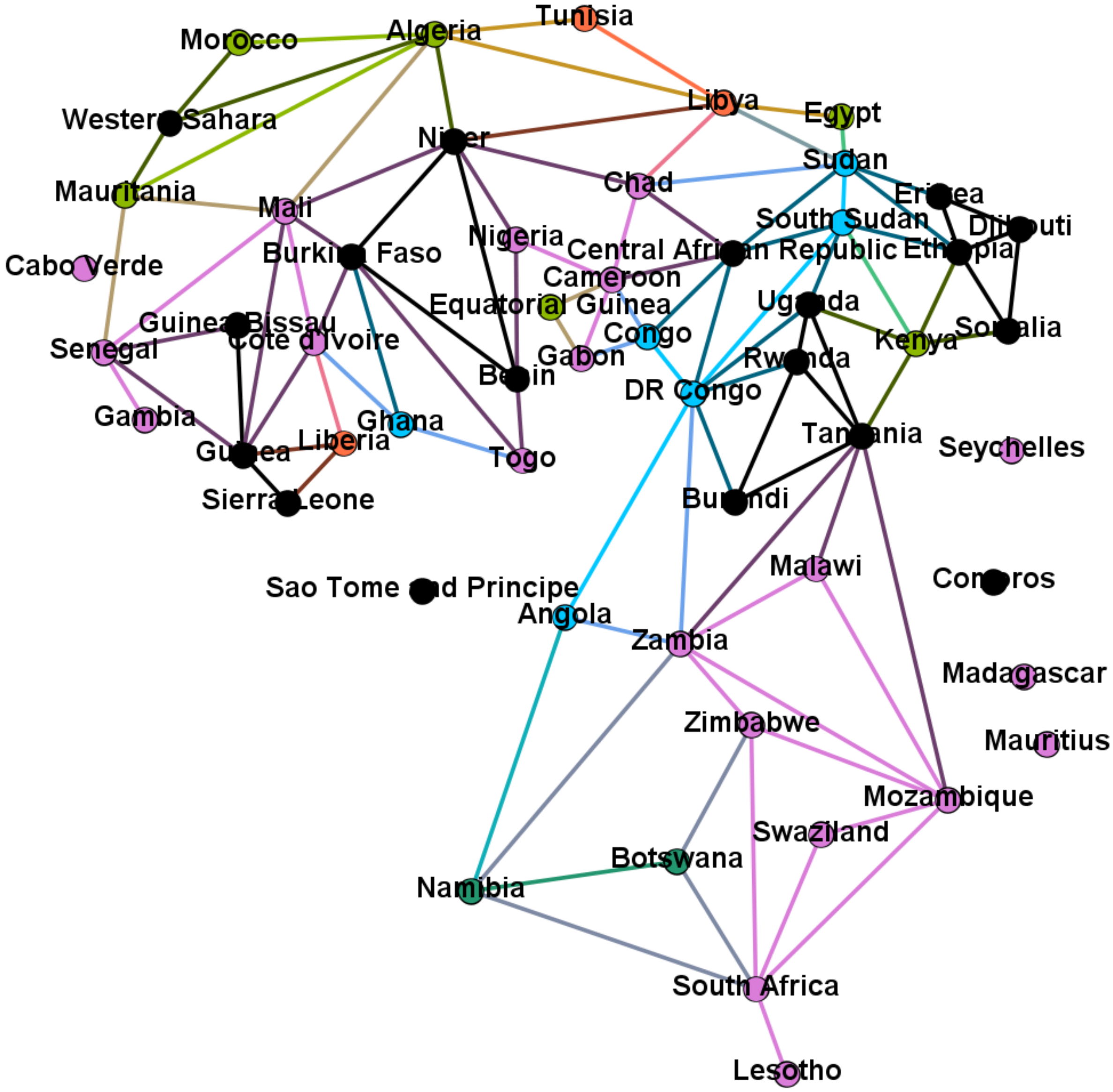


Figure 35: Planar graph, N_{map2}, with the node colouring of N_p

4.1.4 Discussion

Arms trade can fluctuate a lot from year to year [2]. Even though we added records from 2016, in N_{a1}, we obtained consistent results with the network from article [2]. In Figure 31, we see that N_{a1} partitions into three distinct communities. Each community clusters around high degree nodes which coincide with countries that are widely thought of as powerful countries.

The green community is clustered around Russia and China who are both integral actors in international politics. Most of the other nodes in the green community represent Eastern European, Asian, and Central African countries. The blue community is clustered around the U.S. and France. It includes North American, European, Middle Eastern, and North African countries. The orange community is clustered around Germany and it includes mostly European countries.

Network N_{a1} shows that bilateral arms trade is a good indicator of not only bilateral relationships, but also of multilateral ones [2]. The Eastern European and Asian countries in the green community are geographically close to Russia and China, and have historical ties. Moreover, it makes sense that Russia and China are in the same community because they have a shared interest in counter balancing U.S. power. Consequently, it is logical that the U.S. is a high degree node with connection in multiple regions worldwide. In addition, the Netherlands and former Dutch colony, South Africa, have historical ties and are both in the orange group despite not having an edge between them.

In Figure 32, we have network N_{a2} which shows a subgraph of N_{a1}. It has only the top fifteen influential countries discussed in [2] and the African countries that participated in arms trade from 2006 to 2016. This graph shows a more definitive divide between Eastern and Western alliances. One thing that stand out about theses two communities is that the orange community is noticeably more dense than the green one. The orange community has graph density 0.500; whereas, the green community has density 0.170. This may reflect that the power in the green community is more centralized and more decentralized in the orange community.

There are many oil rich countries in Africa and often those country will use the income from oil trade to purchase arms [3]. Figure 33 shows network N_p which represents oil exports from African countries in 2016. It was presumed that there should have been a mirroring of communities between networks N_{a1} and N_p; however, the only evidenced of both arms and oil trade between two countries based on these two graphs is between China and Sudan. In network N_{a1}, we see that China is engaged in arms trade with Sudan; as well as, with the Ukraine which trades with the Democratic Republic of Congo (DR Congo) and Chad. In network N_p, we see that China imports oil from the DR Congo, Chad, Sudan and South Sudan. These African countries are in a particularly violent region, mainly due to civil wars [3]; thus, it would be interesting to explore China's relationship with these countries beyond these two networks. In addition, we notice in Figure 33 that China imports a considerable amount of oil from Angola which is striking because they are neither close in geography, nor is there a commonly known relationship between the two nations outside of trade.

Another noticeable member of the network N_p is the node representing Gabon in purple. In the following, Figure 36, we see Gabon and its connections based on the countries it exports oil to. Notice that it is linked to countries in four out of the five communities in the network.

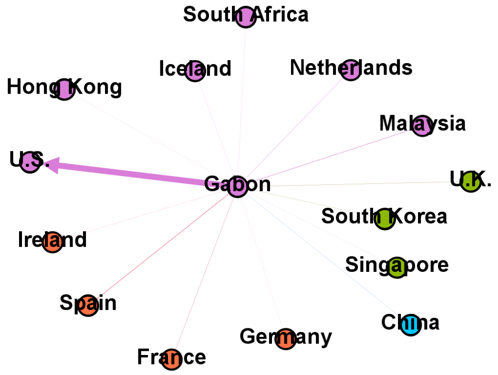


Figure 36: Subgraph of N_p showing Gabon and the countries it exports oil to

Finally, we have Figure 34 and Figure 35, showing a planar graph of the map of Africa coloured based on the communities in networks N_{a1} and N_p, respectively. The two versions of this planar graph are intended to explore the geographical aspect of arms and oil trade. One unexpected result is that Algeria, Tunisia and Libya are all neighbouring Arabic speaking countries; however, Algeria is not in the same community as Tunisia and Libya in either partitioning. In Figure 35, the purple community is spread out on opposite sides of the continent and has little overlap with the communities in Figure 34. Future work could investigate what connects these two regions and their trading partners.

Although we did not get the mirrored community structure between networks N_{a1} and N_p that was expected, the results of these graphs did reveal some relationships that were predictable and some that were surprising. Future work could include exploring some of the less expected relationships; in particular, it would be valuable to look into alliances with Gabon because it has potential of being an important link in other networks.

4.2 The Real Donald Trump

4.2.1 Data Set

It has been estimated that 62 percent of Twitter users live in the U.S., making it an especially impactful social media platform for Americans [28]. Before becoming president of the U.S., Donald Trump had a history of exchanging harsh words over Twitter. Remarkably, he did not change his Twitter habits during his campaign or even once elected president. He continued to use his personal Twitter account, @realDonaldTrump, to share anything that came to his mind. The fixation on his Twitter usage intensified when he tweeted classified information, as well as fired his Secretary of State via his personal account [21, 22].

In this section, we will discuss the linguistic networks drawn from President Trump's use of language on his personal Twitter account. The tweets were obtained from an on-line archive of Trump's tweets [30]. The tweets are stored by year as separate JSON files. Each JSON object includes information, such as date and time, and number of times a tweet was favourited. Here, we are just concerned with the content of each tweet itself. We used tweets from the years 2015, 2016, 2017, and 2018. Using the statistical software R, we eliminated the tweets that Trump retweeted from other accounts because they are not directly from him. Then we eliminated symbols and extra whitespace, as well as made everything lower case for consistency.

4.2.2 Methods

We made four graphs from President Trump's Twitter data, each one using data from the years 2015, 2016, 2017, and 2018, respectively. To allow for reasonable computing time, for each graph we used approximately a quarter of his total number of tweets each year.

The first network, written N_A, is Trump's top one hundred most frequently used words from the last 1884 tweets of 2015. The second network, written N_B, is Trump's top one hundred most frequently used words from the last 1056 tweets of 2016. The third network, written N_C, is Trump's top one hundred most frequently used words from the last 1302 tweets of 2017. Finally, the fourth network, written N_D, is Trump's top one hundred most frequently used word from the first 649 tweets of 2018, which are his available tweets of this year until around April [30]. The decision to use the tweets from the last quarter of the year, as opposed to first, second, or third quarter, was arbitrary.

After removing all of the retweets, we extracted the four networks where each node is one of the top one hundred most used words and there is an edge between two nodes if they are used in the same tweet. The edges are weighted based on how many times two words occur in the same tweet. The top one hundred used words were from a list that was filtered to only contain important words, meaning all smalls words like *the* and *it* were removed. Some nodes were combined if the term was made up of two words; for example, 'white' and 'house' became 'whitehouse'. Moreover, there is often a node representing the singular of a word and another node representing the plural in the same network; for example, 'email' and 'emails' in N_B. We made the choice to keep both nodes and to keep both nodes separate because either the nodes came up in different communities or each node linked their community to two other different communities. To visualize these networks we used the ForceAtlas2 layout algorithm with LinLog mode, *Dissuade Hubs*, Label Adjust, and settings *gravity* and *scaling* equal to 10.0. To analyze the networks we will use the modularity score, average clustering coefficient, average degree, and diameter.

Note that the size of the nodes are proportionate to their degree.

4.2.3 Results

In Figure 37, we have N_A using tweets from 2015. Its modularity score is $Q_{N_A} = 0.222$ and it is partitioned into five communities. The purple community includes the words ‘fox’, ‘carson’, ‘realdonaldtrump’, the green includes ‘kasich’, ‘hillary’, ‘bad’, the orange includes ‘abc’, ‘interviewed’, ‘total’, the teal includes ‘book’, ‘crippled’, ‘live’, and the blue includes ‘support’, ‘post’, ‘south carolina’.

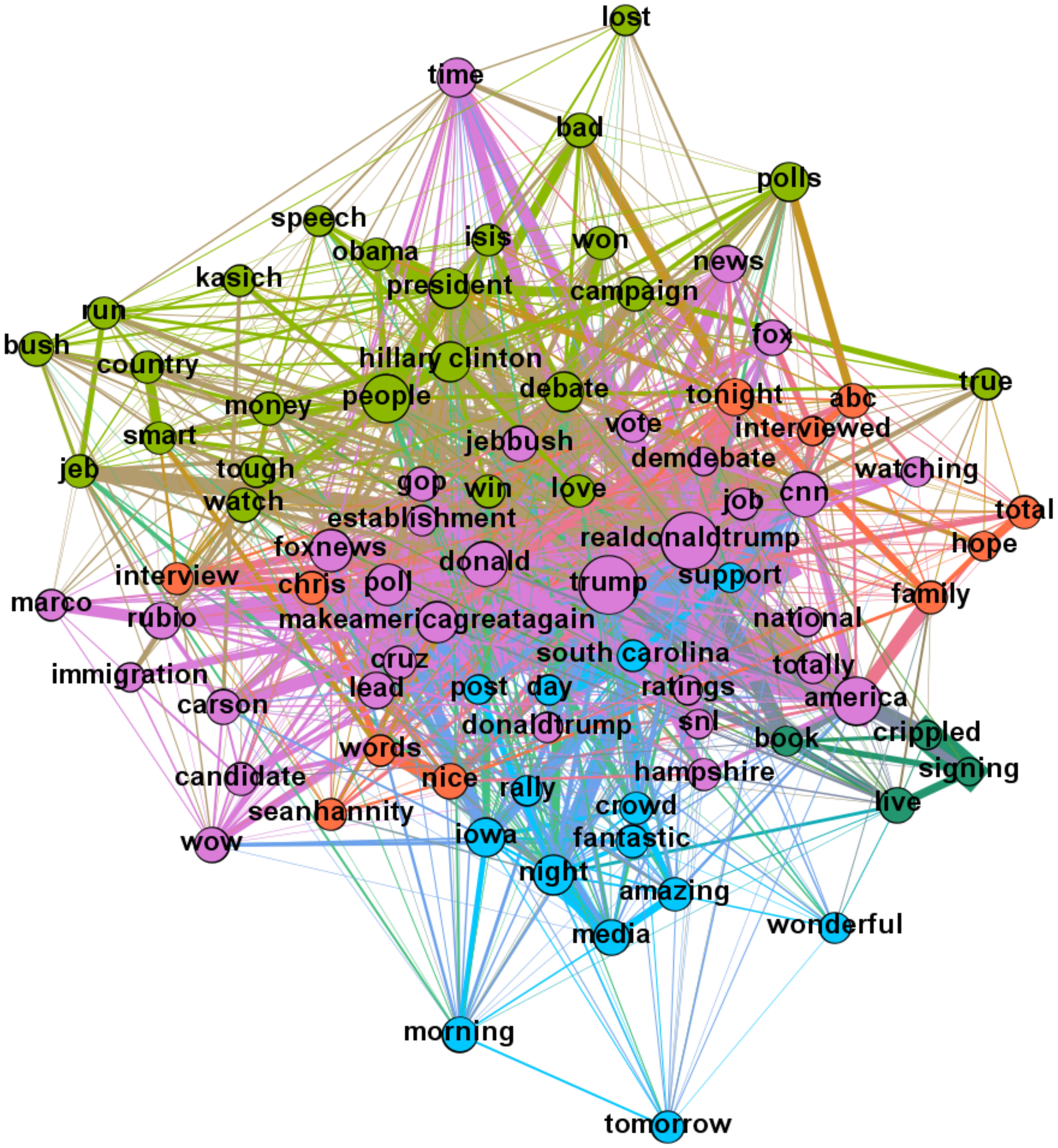


Figure 37: Network N_A of top one hundred used words from the last quarter of 2015.

Next, in Figure 38, we have N_B with modularity score $Q_{N_B} = 0.336$ and is partitioned into five communities. The blue community includes the words 'ohio', 'maga', 'colorado', the orange includes 'pence', 'kaine', 'iowa', the green includes 'watch', 'rally', 'crowd', the purple includes 'campaign', 'women', 'media', and the teal includes only 'polls' and 'john'.

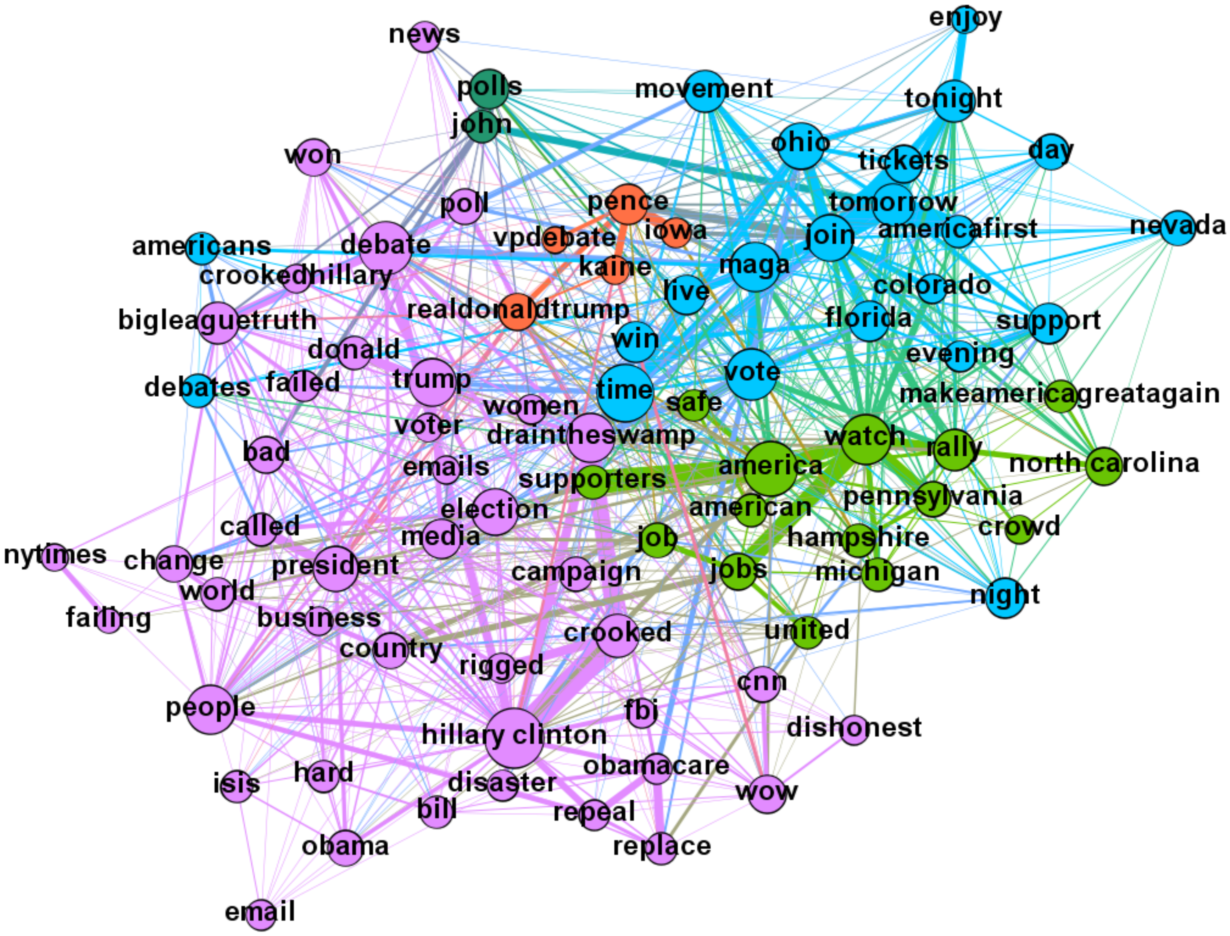


Figure 38: Network N_B of top one hundred used words from the last quarter of 2016.

The third network, N_C in Figure 39, has modularity score $Q_{N_C} = 0.363$ and is partitioned into five communities. The purple community includes the words ‘nfl’, ‘florida’, ‘north korea’, the blue includes ‘happy’, ‘market’, ‘cnn’, the orange includes ‘wonderful’, ‘dems’, ‘crooked’, the green includes ‘cut’, ‘reform’, ‘senate’, and the teal includes ‘crime’, ‘vote’, ‘alabama’.

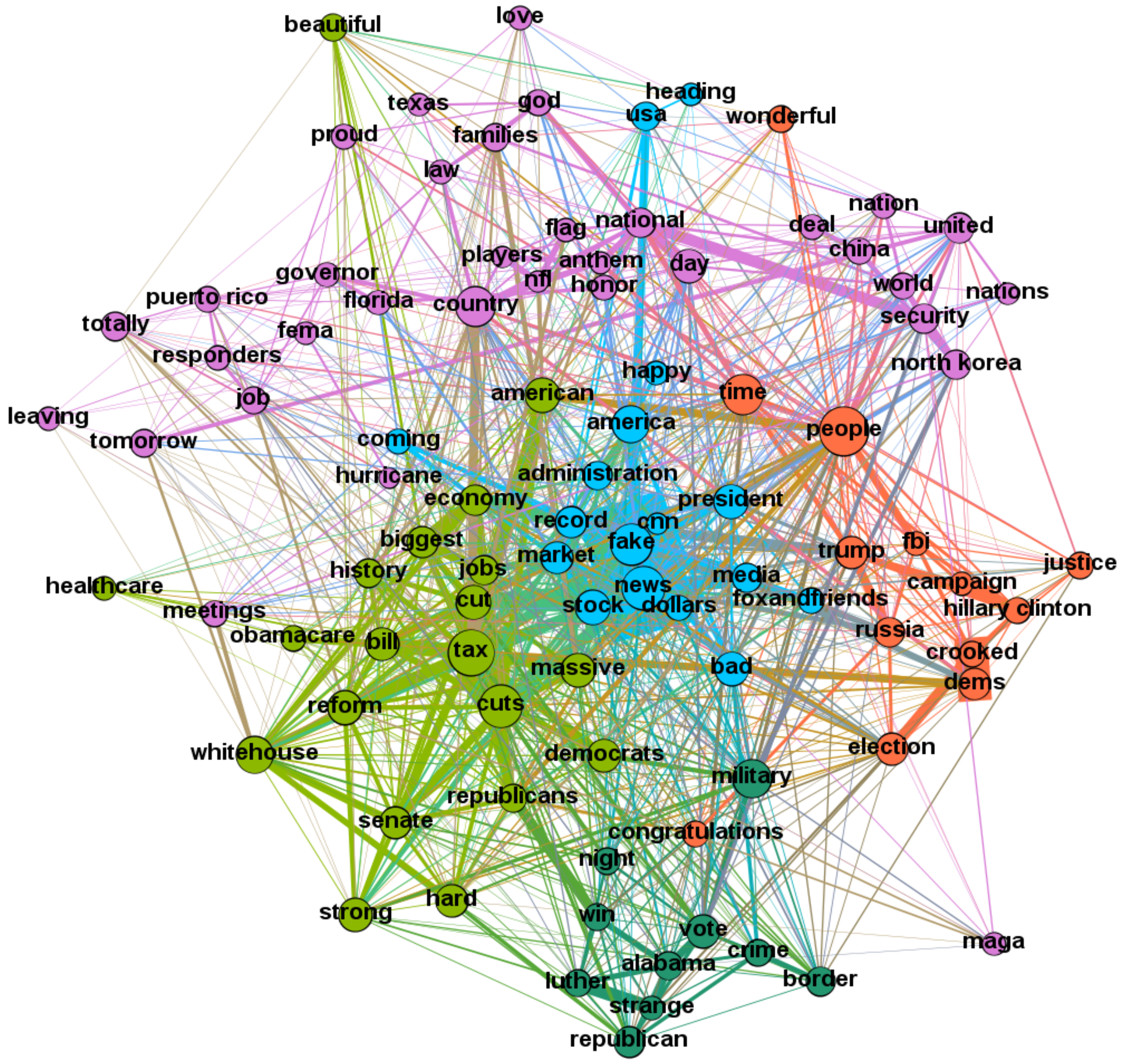


Figure 39: Network N_C of top one hundred used words from the last quarter of 2017.

Finally, in Figure 40, N_D has modularity score $Q_{N_D} = 0.323$ and is also partitioned into five communities. The blue community includes the words ‘trump’, ‘special’ ‘russia’, the orange includes ‘fair’, ‘china’ ‘dollars’, the purple includes ‘massive’, ‘stop’, ‘school’, the teal includes ‘deal’, ‘bill’, ‘dems’, and the green includes ‘jobs’, ‘history’, ‘fbi’.

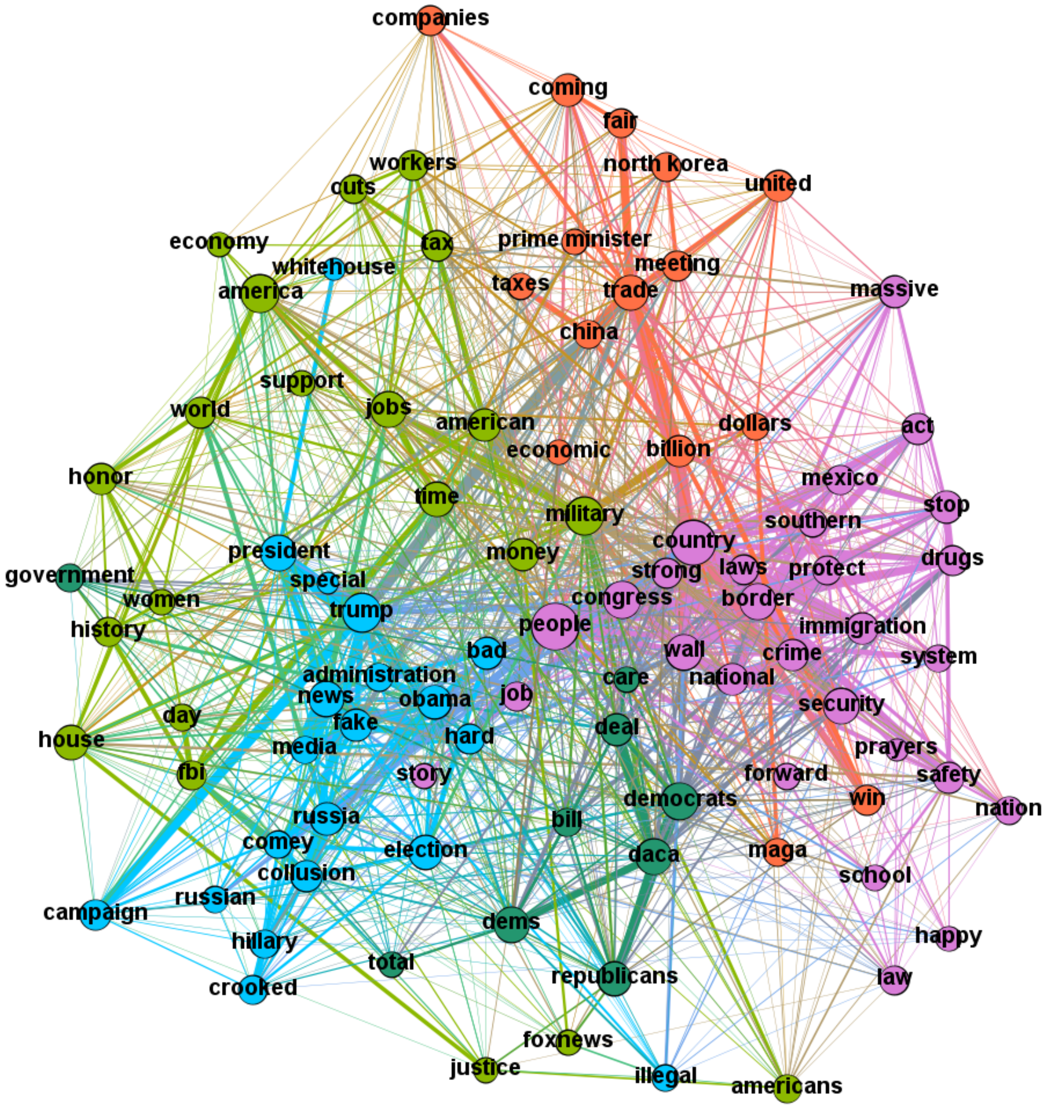


Figure 40: Network N_D of top one hundred used words from all tweets in 2018.

The following table is a summary of the average degree, the diameter, the average path length and the diameter of all four networks.

Network	$\text{deg}_{\text{av}}(G)$	$\text{diam}(G)$	$L(G)$	$C(G)$
N_A	24	3	1.725	0.510
N_B	19.091	3	1.846	0.434
N_C	24.306	3	1.772	0.467
N_D	27.621	3	1.714	0.469

4.2.4 Discussion

Donald Trump is a regular Twitter user. Based off the archived tweets, we know he has been tweeting since around 2009 and tweeting multiple times a day since 2011 [30]. Even though he tweets often, he tends to focus on a handful of topics. The four networks we created follows the time-line starting with the U.S. primary election in 2015, the U.S. general election in 2016, Trump's first year as president in 2017, and Trump's presidency so far for 2018.

In Figure 37, we have network N_A representing Trump's tweet from the end of 2015. The two highest degree nodes are the ones representing 'trump' and 'realdonaldtrump'. In the purple community there are nodes for 'realdonaldtrump', 'donald', 'trump', and 'donaldtrump' which is probably referring to a hastag. The reason that his name and his own Twitter handle would be repeated often is that in 2015 he was campaigning, so he would quote compliments that people would say about him or tweet at him. Notice that there is a heavily weighted edge between his last name 'trump' and his hashtag 'makeamericagreatagain'.

The purple group which includes 'cruz', 'rubio', and 'carson', and the green group which includes ‘kasich’ and ‘bush’ are the communities formed from Trump tweeting about his opponents in the primary elections. Four out of the five groups include exaggerated words that he uses often. The green community contains ‘bad’, the orange contains ‘total’, the purple contains ‘wow’, and the blue contains ‘fantastic’, ‘amazing’, ‘wonderful’.

Next, in Figure 38, we have the tweets from during the general election forming network N_B. Now you see nodes in the orange community for ‘kaine’ and ‘pence’ who were the two vice presidential candidates. Also, the node representing ‘hillary clinton’ has a higher degree than it did in network N_A because she then was his only opponent. In the purple community there are the words ‘crookedhillary’, ‘fbi’, ‘emails’, and his hashtag ‘draintheswamp’. This community represents his tweets against Hillary Clinton and the Democratic Party. In the green community there are the words ‘rally’, ‘newhampshire’ and ‘florida’, with his hashtag ‘makeamericagreatagain’. Also, in the blue community there are the words ‘ohio’, ‘colorodo’, ‘support’, with his shortened hashtag ‘maga’. These two communities arose from him tweeting to announce his rallies.

In Figure 39, we have network N_C which includes tweets from Trump’s first year as president in 2017. In blue there are nodes representing ‘fake’, ‘news’, ‘cnn’, ‘bad’, ‘media’ and the hashtag for the television show ‘foxandfriends’. Previously, there were media related nodes, such as ‘cnn’ and ‘fox’, throughout the networks N_A and N_B. However, in N_C there is a distinct community related to Trump’s dislike of the media and there is the emergence of the word ‘fake’. We also still have a community with ‘hillary clinton’, ‘fbi and ‘crooked’ in orange. Moreover, he continues to use his slogan *Make America Great Again*, but now on his Twitter he is mainly use the shortened hashtag ‘maga’.

What is notable about this network are that there are more communities surrounding policy. The green community includes the words ‘healthcare’, ‘economy’, ‘jobs’, ‘tax’, ‘reform’, and ‘cut’. This community pertains to domestic issues; whereas, the purple community has a cluster related to foreign issues because it has the words ‘security’, ‘china’, and ‘north korea’. There is also a cluster in the purple community that seem to have a patriotic theme because it includes words such as ‘god’, ‘anthem’, and ‘nfl’.

Lastly, we have the network N_D in Figure 40. These are his tweets from 2018. Once again, we still have a community dedicated to ‘fake’, ‘news’, and ‘media’; as well as, a community dedicated to ‘crooked’ and ‘hillary’, but this time they are all together in the blue community. He also still frequently tweets about ‘tax’, ‘cuts’, and ‘jobs’ in the green community. However, now his themes of foreign issues and patriotism seem to be less definitive. In purple there are the words ‘security’, ‘border’, and ‘wall’. In purple there are the words ‘north korea’, ‘china’, and ‘trade’. The node for ‘russia’, ‘collusion’, and ‘media’ is in blue. The themes of communities from network N_C now find themselves rearranged differently in the communities from network N_D. This suggests that the way he views these themes is changing.

Considering Donald Trump tweets multiple times a day, it is remarkable that out of the most frequently used words from each batch of tweets exactly five communities are detected using the modularity score. Some of the topics reflected the change from before becoming president to after becoming president and there seems to be a change in sentiment based on the difference in communities from 2017 to 2018. However, in all four networks he consistently mentioned the media and Hillary Clinton.

There is a lot of opportunity for future research, both in regards to Donald Trump’s tweets or any other prominent figure’s tweets. Here we focused on themes derived from linguistic networks, but another angle would be to further investigate the structure of his sentences on Twitter. The table with the results for the average degree, diameter, average path length, and average clustering coefficient shows that these metrics are fairly consistent across the four networks. Thus, it would be valuable to analyze his sentence structure using linguistic networks.

5 Conclusion

5.1 Summary of Thesis

Advancements in computing power and increased accessibility to open data have expanded the scope of interdisciplinary research. In particular, they have promoted the intersection of network and social sciences. In this thesis, we discussed various politically themed networks through a graph theory lens. We focused on the aspect of community structure when visualizing real-world networks, using graph theory statistics such as modularity, clustering coefficient, and centrality.

First, we defined what is a graph and introduced basic concepts in graph theory. Then, we extended the conversation to complex networks and detailed their main properties, such as power law distribution, high clustering coefficient, and small world property. In Chapter 2, we explained a common approach to visualizing networks. Here we described the motivations behind force-directed layout algorithms and gave examples of the default algorithm used in the network visualization software Gephi.

Subsequently, we used on-line sources to mine data and extract real-world networks. Chapter 3 contained networks related to Canadian politics. We created three networks based on the voting patterns of Canadian MPs, where each network represented different parliamentary sessions. We observed that the community structure forms along party lines and along possible alliances. We also created a dynamic network from the voting patterns of Toronto city councillors during Rob Ford's term as mayor, where we able to visualize the his core support.

Chapter 4 comprised of networks related to international politics. We recreated a network from [2] using records of international arms trade. Then, we compared it to a network created from records of oil exports from Africa and investigated the role of geography, using a planar graph representing the map of Africa. Here we recognized some known political alliances; as well as, noticed some unexpected connections. Lastly, we sampled tweets from U.S. President Donald Trump's personal Twitter account, @realDonaldTrump, from the years 2015 to 2018, respectively. We established his top one hundred most used words and how often they occurred in the same tweet. We then created four networks each exhibiting a structure based on the reoccurring themes of his tweets.

5.2 Future Work

Each of the networks we studied provide insight into the validity of using network science to analyze themes in political science. Visualizing complex networks serves as a tool to highlight the emergence of hidden community structures, which leads to more meaningful understanding of each network. Here we primarily relied on the modularity scores and clustering coefficients to obtain our results; however, there remain many open problems.

1. The network of Canadian MPs revealed that the community structure based on voting patterns form along the political parties, even when considering as little as three votes. Would the same phenomena be observed if an increasing number of votes were considered? Moreover, we only considered the votes that took place in the House of Commons. What would occur if we considered the MPs' voting patterns of subcommittee votes?

2. As stated in Chapter 3, Toronto city councillors do not have formal political affil-

iations. However, from our results, we see that their voting patterns tend to be consistent. We only considered twenty votes from a particular time period. What community structure would arise if we considered more Toronto city council votes? Do their voting patterns reveal anything about their personal ideologies?

3. In Chapter 4, one of our results from the arms and oil trade analysis was that there was a distinct divide between the countries that traded with Russia and China, and those that traded with the U.S. and Germany. The community surrounding Russia and China seems to exhibit more centralized power than the community containing the U.S. and Germany. Is this phenomena consistent in other trade networks? Also, aside from arms and oil, does geography play a role in other trade alliances?

4. We noticed that Gabon was connected to countries in four out of the five groups in the network. What does this say about Gabon's role in the international community? Do these connections translate into alliances in other networks?

5. When analyzing the networks derived from Donald Trump's personal Twitter account, we focused on the reoccurring themes that arose as clusters in the network. What would a linguistic network focussing on his sentence structure say about his use of Twitter? Moreover, would a similar analysis of other prominent political figures Twitter accounts give the same results?

Appendix

The following is the code used in R to create an edge list from the voting records of Canadian MPs. The same code was used for each of the networks representing different parliamentary sessions; however, the file names were changed accordingly.

```
#40th parl 3rd session: votes 3, 4, 6
v3<-read.csv(’C:/Users/Lyndsay Roach/Documents/40_vote3.csv’)
v3$Name<-as.character(v3$Name)
v3$Label<-as.character(v3$Label)
v4<-read.csv(’C:/Users/Lyndsay Roach/Documents/40_vote4.csv’)
v4$Name<-as.character(v4$Name)
v4$Label<-as.character(v4$Label)
v6<-read.csv(’C:/Users/Lyndsay Roach/Documents/40_vote6.csv’)
v6$Name<-as.character(v6$Name)
v6$Label<-as.character(v6$Label)

#make unique list of labels
L<-rbind(v3[,1:2],v4[,1:2],v6[,1:2])
labes<-unique(L)

write.csv(labes,"G5_40_labels.csv")
#make node list with political affiliation as labels
n<-read.csv(’C:/Users/Lyndsay Roach/Documents/G5_40_labels.csv’)
```

```
n$Name<-as.character(n$Name)
n$Label<-as.character(n$Label)


#changes names into ID numbers
for(i in(1:length(v3$Name))){
  for(j in (1:length(n$ID))){
    if (v3$Name[i]==n$Name[j])
      v3$Name[i]<-n$ID[j]
  }
}


for(i in(1:length(v4$Name))){
  for(j in (1:length(n$ID))){
    if (v4$Name[i]==n$Name[j])
      v4$Name[i]<-n$ID[j]
  }
}


for(i in(1:length(v6$Name))){
  for(j in (1:length(n$ID))){
    if (v6$Name[i]==n$Name[j])
      v6$Name[i]<-n$ID[j]
  }
}


#make edge list

#use package gtools


#vote 3, 4, 6 "Yea"

v3TempY<-v3[ which(v3$Vote.3==1), ]

c3Y<-combinations(length(v3TempY$Name),2, v3TempY$Name, set=FALSE,

repeats.allowed=FALSE)


v4TempY<-v4[ which(v4$Vote.4==1), ]

c4Y<-combinations(length(v4TempY$Name),2, v4TempY$Name, set=FALSE,

repeats.allowed=FALSE)


v6TempY<-v6[ which(v6$Vote.6==1), ]

c6Y<-combinations(length(v6TempY$Name),2, v6TempY$Name, set=FALSE,

repeats.allowed=FALSE)


#vote 3, 4, 6 "Nay"

v3TempN<-v3[ which(v3$Vote.3==0), ]

c3N<-combinations(length(v3TempN$Name),2, v3TempN$Name, set=FALSE,

repeats.allowed=FALSE)
```

```
v4TempN<-v4[ which(v4$Vote.4==0), ]
c4N<-combinations(length(v4TempN$Name),2, v4TempN$Name, set=FALSE,
repeats.allowed=FALSE)

v6TempN<-v6[ which(v6$Vote.6==0), ]
c6N<-combinations(length(v6TempN$Name),2, v6TempN$Name, set=FALSE,
repeats.allowed=FALSE)

e40YN<-rbind(c3Y,c4Y,c6Y,c3N,c4N,c6N)

#add weights
#data.table
count.dups <- function(DF){

  DT <- data.table(DF)
  DT[,.N, by = names(DT)]
}

e40YN<-count.dups(e40YN)

#export edge list
write.csv(e40YN,'G5_e40YN.csv')
```

The following is the code used in R to clean and mine Twitter data from @realDonaldTrump; as well as, create an edge list. The same code was used for each of the networks representing tweets from different years; however, the file names were changed accordingly.

```
#using packages jsonlite, dplyr, tidytext
tweets<-fromJSON('C:/Users/Source/Documents/condensed_2018.json
',flatten=TRUE)
text<-tweets$text
text2<-text#[1:10]
#drop retweets, run until all rts are droppped
for(i in (1:length(text2))){
  if(grepl("^RT", text2[i], perl=TRUE)=='TRUE'){
    text2<-text2[-c(i)]
  }
}


#remove symbols
temp<-stringr::str_replace_all(text2,"[^a-zA-Z\\s]", " ")
#reduce whitespace
temp2<-stringr::str_replace_all(temp,"[\\s]+", " ")
#make everything lower case
temp3<-tolower(temp2)
```

```
#split string into a list of tweets and split words in each
tweet
tst<-strsplit(temp3, '\\W+', perl=TRUE)
#store it as a list of each word then store list as vector
tst2<-unlist(tst, recursive = TRUE, use.names = TRUE)

df<-data_frame(text2)
mystopwords<-data_frame(word=c("https","t.co","amp","rt"))
#remove stopwords
tmp<- df %>%
  unnest_tokens(word, text2) %>%
  anti_join(stop_words)
df_tidy<-tmp%>%
  anti_join(mystopwords,by="word")
#determin frequency of words, outputs in descending order
freq<-df_tidy %>%
  count(word, sort = TRUE)

#choose list of top most used words
freq2<-freq$word
bank<-freq2[1:100] #list of top most used words

#make table of top words and the tweet number and the index
```

```
of that

word in the tweet

v1<-vector()

v2<-vector()

v3<-vector()

for(i in (1:length(tst))){

  for (j in (1:length(tst[[i]]))){

    if (tst[[i]][j] %in% bank){

      v1<-append(v1,tst[[i]][j])

      v2<-append(v2,i)

      v3<-append(v3,j)

    }


  }

}


w2<-as.numeric(v2)

w3<-as.numeric(v3)

#positions of each top 100 words in tweets

Lt<-cbind(v1,w2,w3)


#empty vectors to store edge list

source<-vector()
```

```
target<-vector()
vec<-vector()
#make a list of the words that appear within the same tweet
for(i in (1:length((w2)-1))){
  for(j in ((i+1):length(w2))){
    if ((w2[i])==(w2[j])){
      vec<-append(vec,w2[i])
      source<-append(source,w3[i])
      target<-append(target,w3[j])
    }
  }
}

#remove loops
L<-vector(mode=’numeric’,length=length(vec))
L2<-data.frame(cbind(vec,source,vec,target,L))

for (i in (1:length(vec))){
  if(tst[[vec[i]]][source[i]]==tst[[vec[i]]][target[i]]){
    L2$L[i]<-L2$L[i]+1
  }
}
```

```
}

L3<-L2[which(L2$L==0),]

L3<-L3[,1:4]


#add weights

w<-rep(1,length(L3$vec))

L4<-cbind(L3,w)


for (i in (1:(length(L3$vec)-1))){

  for (j in (i+1):length(L3$vec)){

    if((tst[[vec[i]]][source[i]]==tst[[vec[j]]][source[j]]) &&

    (tst[[vec[i]]][target[i]]==tst[[vec[j]]][target[j]])){

      L4$w[i]<-L4$w[i]+1

    }

  }

}


for (i in (1:(length(L3$vec)-1))){

  for (j in (i+1):length(L3$vec)){

    if((tst[[vec[i]]][source[i]]==tst[[vec[j]]][source[j]]) &&

    (tst[[vec[i]]][target[i]]==tst[[vec[j]]][target[j]])){

      L4$w[j]<-0

    }

  }
```

```
}


L5<-L4[which(L4$w!=0),]


#making csv files with edge list

s<-vector()

t<-vector()

nodes<-freq$word


for (i in (1:length(L5$vec))){

  if (tst[[L5$vec[i]]][L5$source[i]] %in% nodes)

    s<-append(s,which(nodes==tst[[L5$vec[i]]][L5$source[i]]))

}


for (i in (1:length(L5$vec))){

  if (tst[[L5$vec[i]]][L5$target[i]] %in% nodes)

    t<-append(t,which(nodes==tst[[L5$vec[i]]][L5$target[i]]))

}


w<-L5$w

edges<-cbind(s,t,w)


#write into csv file
```

```
write.csv(nodes,"Trump_2018_nodes.csv")

write.csv(edges,"Trump_2018_edges.csv")
```

References

[1] Africa Map http://ontheworldmap.com/africa/

[2] Algobeans https://algobeans.com/2016/04/12/network-graphs-where-will-your-country-stand-in-world-war-iii/

[3] Mark Anderson, Achilleas Galatsidas, Global weapons trade targets Africa as imports to Algeria and Morocco soar, *The Guardian* (2015).

[4] Anonymous, Gabon, *Political Science Database: SciTech Premium Collection* (2012).

[5] N.L. Biggs, E. K. Lloyd and R. J. Wilson, *Graph Theory: 1736-1936*, Oxford University Press Inc., 1986.

[6] Nicholas J. Caruana, R. Michael McGregor, Aaron A. Moore, Laura B. Stephenson, Voting 'Ford' or Against: Understanding Strategic Voting in the 2014 Toronto Municipal Election, *Social Science Quarterly* **99** (2017) 231–245.

[7] F. Chung and L. Lu, *Complex Graphs and Networks*, American Mathematical Society, 2006.

[8] Phillip E. C. Compeau, Pavel A. Pevzner, and Glenn Tesler, Why Are de Bruijn Graphs Useful for Genome Assembly? *Nature biotechnology* **11** (2011) 987–991.

[9] Councillor Voting Records https://torontoist.com/2011/09/fordiness-chart/

[10] Ben Dummett, Toronto City Council Urges Mayor to Step Aside; Ford Apologizes for Drug Use, but Resists Nonbinding Motion, *Wall Street Journal* **Online** (2013).

[11] Ben Dummett, Toronto's City Council Strips Mayor of More Powers; Ford Remains Defiant, Calling Vote a 'coup d'état' and Vowing Court Challenge, *Wall Street Journal* **Online** (2013).

[12] Rand Dyck and Chistopher Cochrane, *Canadian Politics: Critical Approaches*, Nelson Education Ltd., (2014).

[13] K. Erciyes, *Complex Networks: An Algorithmic Perspective*, CRC Press Taylor and Francis Group, 2015.

[14] T. M. J. Fruchterman and E. M. Reingold, Graph Drawing by Force-directed Placement, *Software: Practice and Experience* **21** (1991) 1127–1164.

[15] Jean-François Godbout and Bjørn Høyland, Legislative Voting in the Canadian Parliament, *Canadian Journal of Political Science* **44:2** (2011) 367–388.

[16] House of Commons `https://www.ourcommons.ca/`

[17] Yifan Hu, Efficient and high quality force-directed graph drawing, *The Mathematica Journal* **10** (2006) 37–71.

[18] Y. Hu and L. Shi, Visualizing large graphs, *WIREs Comput Stat* **7** (2015) 115–136.

[19] M. Jacomy, T. Venturini, S. Heymann, M. Bastian, ForceAtlas2, a Continuous Graph Layout Algorithm for Handy Network Visualization Designed for the Gephi Software, *PLOS One* **6** (2014).

[20] M. Jacomy and S. Heymann, Gephi : An Open Source Software for Exploring and Manipulating Networks, *Association for the Advancement of Artificial Intelligence* (2009).

[21] Sarah Jeong, Trump fires secretary of state via tweet, *The Verge* (2018).

[22] Ken Meyer, Mediaite: Remember When Trump Was Concerned About Classified Information Leaks? Twitter Does, *American Psychological Association* **6** (2017).

[23] Amy N. Langville, Carl D. Meyer, Deeper Inside PageRank, *Internet Mathematics* **1** (2004) 335–380.

[24] Carlo Dal Maso, Gabriele Pompa, Michelangelo Puliga, Gianni Riotta, Alessandro Chessa, Voting Behavior, Coalitions and Government Strength through a Complex Network Analysis, *PLOS One* **9(12)** (2014).

[25] A. Noack, Energy Models for Graph Clustering, *Journal of Graph Algorithms and Applications* **11** (2007) 453–480.

[26] H-W.Shen, *Community Structure of Complex Networks*, Springer Berlin Heidelberg, 2013.

[27] Stockholm International Peace Research Institute `http://armstrade.sipri.org/armstrade/page/trade_register.php`

[28] Tamara A. Small, What the Hashtag? A content analysis of Canadian politics on Twitter, *Information, Communication and Society* **14** (2011) 872–895.

[29] Trade Map `https://www.trademap.org/`

[30] Trump Tweet Data Archive `https://github.com/bpb27/trump_tweet_data_archive`

[31] D. J. Watts, *Small Worlds: The Dynamics of Networks between Order and Randomness*, Princeton University Press, 1999.

[32] Baoyindureng Wu and Wanping Zhang, Average distance, radius and remoteness of a graph, *Ars Mathematica Contemporanea* **7** (2014) 441–452.