OPTIMAL TIME INTERVALS DETECTION FOR DYNAMIC NETWORKS

(DİNAMİK AĞLAR İÇİN OPTİMAL ZAMAN DİLİMİ TESPİTİ)

by

Nadir Türe

Thesis

Submitted in Partial Fulfillment

of the requirements

for the degree of

MASTER OF SCIENCE

in

COMPUTER ENGINEERING

in the

INSTITUTE OF SCIENCE AND ENGINEERING

of

GALATASARAY UNIVERSITY

Supervisor: Asst. Prof. Dr Keziban Orman

GALATASARAY UNIVERSITY

April 2021

This is to certify that the thesis entitled

OPTIMAL TIME INTERVALS DETECTION FOR DYNAMIC NETWORKS

prepared by **Nadir TÜRE** in partial fulfillment of the requirements for the degree of **Master of Science in Computer Engineering** at the **Galatasaray University** is approved by the

Examining Committee:

Assist. Prof. Dr. G. Keziban ORMAN (Supervisor)
Department of Computer Engineering
Galatasaray University ----------

Assist. Prof. Dr. B. Atay ÖZGÖVDE
Department of Computer Engineering
Galatasaray University ----------

Prof. Dr. Selim BALCISOY
Department of Computer Science & Engineering
Sabancı University ----------

Date: ----------

ACKNOWLEDGEMENTS

I would like to express my gratitude to my thesis supervisor Asst. Prof. Dr Keziban Orman for her unlimited support and guidance.

I would also like to thank our collaborators from Sabancı University, Dr. Selim Balcısoy and Hasan Alp Boz.

Last but not the least, I would like to thank my family, my parents, my brother and my sister.

Table of Contents

List of Figures

List of Tables

ABSTRACT

Extracting a proper dynamic network for modelling a time-dependent complex system is an important issue for analyzing the studied system. Building a correct model is related to finding out critical time points where a system exhibits considerable change. This task is usually executed by a comparative analysis of time series reflecting the features of extracted dynamic networks. These time series are generated by sliding different sizes windows at the entire time span of system. The most commonly used features are the networks' topological properties. In this work, we not only look for topological properties evolution but also propose to measure network stability to detect proper time intervals. We develop three similarity scores to measure node, link, and neighborhood similarities of any consecutive snapshots of a dynamic network. We consult to their statistically expected values which are calculated under the null model of proposed metrics in order to determine whether the network stays stable. We validate the usability of the proposed metrics and of their expected values by demonstrating their reaction to different window sizes. We use two different data sets having different temporal dynamics. First one is an original data set which is log data collected from the Wi-Fi access points in Sabancı University Campus and second one is Enron emails. According to the results, the proposed similarities help to distinguish critical time intervals more effectively than previously proposed metric. Because there is an objective comparison thresholds, i.e. their statistically expected values, proposed similarities are system and methodology independent.

ÖZET

Zamana bağlı karmaşık bir sistemi modellemek için uygun bir dinamik ağ çıkarmak, incelenen sistemi analiz etmek için önemli bir konudur. Doğru bir model oluşturmak, bir sistemin önemli ölçüde değişiklik gösterdiği kritik zaman noktalarını bulmakla ilgilidir. Bu işlem, genellikle oluşturulan dinamik ağlarının özelliklerini yansıtan zaman serilerinin karşılaştırmalı analizi ile yapılır. Bu zaman serileri sistemdeki bütün zaman aralığı içinde farklı boyutlardaki zaman aralıkları ile oluşturulur. En sık kullanılan özellikler bu ağların topolojik özellikleridir. Bu çalışmada, yalnızca topolojik özelliklerin gelişimini araştırmakla kalınmadı, aynı zamanda uygun zaman aralıklarını tespit etmek için ağ kararlılığını ölçmeyi önerildi. Ardışık anlık ağlar arasında düğüm, bağlantı ve komşuluk benzerliği olacak şekilde üç benzerlik skoru geliştirdi. Ağın kararlı kalıp kalmayacağını belirlemek için önerilen metriklerin sıfır modeli altında hesaplanan, istatistiksel olarak beklenen değerlerine bakıldı. Önerilen metriklerin ve beklenen değerlerinin kullanılabilirliğini, farklı pencere boyutlarına tepkilerini göstererek doğrulandı. Farklı zamansal dinamiklere sahip iki farklı veri seti kullanıldı. Birincisi, Sabancı Üniversitesi Kampüsü'ndeki Wi-Fi erişim noktalarından toplanan log verilerinden derlenen ve daha önce kullanılmayan orijinal veri seti, ikincisi ise Enron e-postalarıdır. Sonuçlara göre, önerilen benzerlikler, kritik zaman aralıklarını daha önce önerilen metrikten daha etkili bir şekilde ayırt etmeye yardımcı olduğu gözlendi. Bu sonuca erişilebilmesinin sebebi, istatistiksel oalrak beklenen değerler ve önerilen benzerlikler, sistem ve metodolojiden bağımsızdır.

1 INTRODUCTION

Analysis of dynamic networks has been attracted to the attention of many scientists from different research domains (Kuhn & Oshman, 2011; Spiliopoulou, 2011; Blonder et al., 2012; Holme & Saramäki, 2012; Aggarwal & Subbian, 2014; Leskovec et al., 2005, 2007; Kempe et al., 2000; Rossi et al., 2013; Berger-Wolf & Saia, 2006). Dynamic representation provides richer modelling than static complex networks because it takes possible temporal changes of actors and their interactions into account. There are several different representations to build dynamic networks such as event lists, link streams, or event-based sequential graphs (Holme & Saramäki, 2012). One of them is a time-based sequential graph representation of complex systems whose components change over time. Each member of the network sequence is called a *snapshot*. Snapshots contain interactions between the observed system actors for a given *time interval*. In this work, we focus on proper dynamic network extraction for time-based sequential graph modelling.

Proper dynamic network extraction of a system is a challenging issue because it is related to making decisions on the model components such as nodes, links or time intervals for snapshots. Usually, the network nodes and their links represent the actors of the system and their interactions, respectively. Those components are according to the need of an application, which in turn no need of specific methodology. However, which time intervals to select for extracting snapshots is an open subject. This selection directly determines the topology of the resulting network sequence. In (Rocha et al., 2017), the authors concentrate on this problem. They underline that time interval directly affects the dynamics of simulated epidemics and information spread, mixing properties of random walk, and synchronization on networks. They study the effects of total observation time of system, the selected time intervals, and the number of nodes and links in the snapshots on the statistics that represent the properness of final network. Since the structure of the model will affect all further analyzes, i.e., community detection

(Medo et al., 2018; Dakiche et al., 2018), link prediction, attribute prediction, and change-point detection (Fish & Caceres, 2017), etc., it is essential to choose proper time intervals for snapshots extraction. Recently, we come across to the studies which are dedicated on *finding proper time intervals* (Zhang et al., 2012; Fish & Caceres, 2017; Medo et al., 2018; K. Darst et al., 2016; Soundarajan et al., 2016; Uddin et al., 2017; Caceres et al., 2011; Dakiche et al., 2018). Accordingly, the system's time span should be divided into time intervals that are neither small enough to make the network noisy nor large enough to ignore significant time-dependent effects on the network (Fish & Caceres, 2015; Soundarajan et al., 2016; Sulo et al., 2010). In previous studies, we see that finding a proper time interval by respecting these two limits is not defined as an independent problem but defined according to the analysis method used in the application. An empirical procedure is widely used in the literature (Sulo et al., 2010; Uddin et al., 2017; Krings et al., 2012; Fish & Caceres, 2015; K. Darst et al., 2016; Fish & Caceres, 2017; Clauset & Eagle, 2012). First, the authors divide the time span of their system into different time intervals based on the window sizes they had previously determined and extract candidate dynamic networks containing snapshots generated according to each time interval. Second, they generate time series of snapshots' features for each candidate dynamic network. Third, they make the decision of proper time intervals according to the robustness of the generated time series.

This study's primary purpose is to propose a formal definition of the proper dynamic network extraction problem. We aim to propose a generic problem definition to overcome two critical issues that appear in previous studies. First, most of the previous works use topological properties, i.e., average path length, radius or density, etc., of snapshots as the features for time series generation (Sulo et al., 2010; Uddin et al., 2017; Soundarajan et al., 2016). However, although the network structure changes considerably, topological properties' values may stay the same or close for its snapshots. Topological properties do not reflect the exact changes in network components, i.e., nodes, links. They summarize the network structure. Moreover, most of them are not normalized metrics. More clearly, their values depend on the topological structure of the studied snapshot. One cannot use their scores for comparing different snapshots. Although examining their time series gives an intuition about the proper time intervals for the system, their usage does not constitute a generic approach independent of system properties. We come across the usage of snapshots' spectral properties (Clauset & Eagle, 2012) or similarity

metrics (K. Darst et al., 2016) for overcoming this problem. Second, most of the previous works focus on the time series analysis methodology. We come across that the robustness of the generated time series is stated by either statistical time series analysis (Sulo et al., 2010) or unsupervised (Uddin et al., 2017) or supervised learning (Fish & Caceres, 2017). Supervised learning approaches demand labelling subjectively the training data set. Unsupervised learning produces the results which are already subjective. Most of those approaches divide the time span into regular time intervals. Nevertheless we also come across the works that find time intervals with different duration (Soundarajan et al., 2016). This division uses an event-based approach, i.e., uses small periods of time when many events that occur, and large periods otherwise (Soundarajan et al., 2016). Whether they find regular or variable time intervals, the generated time series's nature may not be suitable for the specified analysis methods. Hence, they cannot be generalized. Therefore; there is a need for a method-independent formal definition for the problem.

In this work, we define the finding proper time intervals as a problem of partition the continuous-time span according to the time points at which the system's stability is broken. We concentrate on two critical issues; (1) measuring the stability of the system by taking into account all its components and (2) making an objective decision of system stability breaks. For the first point, we developed three metrics, a.k.a node, link, and neighborhood similarities, based on well-known Jaccard Similarity in order to measure consecutive snapshots' stability. Those metrics reflect the proportion of the components which are not changing for two consecutive snapshots. Those proportions are computed among the overall components of consecutive snapshots. If the snapshots are entirely different than metrics take zero. If they are entirely the same, their value becomes one. For the second point, rather than using a specific analysis method or a predefined threshold, we consult statistical randomness limits of Jaccard Similarity. We propose to compare the result similarity score with the one generated for its null model. At that point, we consult a null model, which is proposed for Jaccard Similarity(Real & Vargas, 1996). This null model uses the sizes of the compared sets to determine whether they are similar by chance. The time intervals whose similarity scores are lower than these limits are the critical time intervals where the system change is greater than chance. Thus, using the null model helps us to make objective decisions.

We validate the usability of proposed metrics with their null models on two different

data sets, which have very different temporal change trends. The first data set is Wi-Fi access points (WAP) logs of Sabancı University Tuzla Campus, Istanbul. This data set is an original data set that has not been analyzed before this study. There are many problems limiting the handling of raw WAP logs. In (Kjærgaard & Nurmi, 2012), the authors explain in detail the challenges of using WAP logs for extracting social information. One needs to model this data accurately for overcoming these limitations. Here, we use it as a case study for our problem statement. The second set that we have used is Enron data set (Sulo et al., 2010; Fish & Caceres, 2017; K. Darst et al., 2016; Clauset & Eagle, 2012). It is a well-known data that is modelled both as static and dynamic networks. We compare the qualitative performance of proposed similarity metrics with Clauset's adjacency correlation coefficient (Clauset & Eagle, 2012) which is another metric developed for same issue.

The main contributions of this work are three folds. First, we propose an application and method independent definition for proper dynamic network extraction problem. It is based on measuring systems' stability objectively. Second, we propose three new network similarity metrics. We compare their scores with the one calculated for their null model. We use them as objective criteria for finding critical time points that the system exhibits a considerable change. Third, we validate our proposal on two data sets: Enron email and WAP logs of Sabancı University. The readers find the details of the data set descriptions in section 3 and problem definition and proposed methodology in section 4. We give empirical results in section 6. Afterwards, in section 7, we summarize the overall study and discuss its perspectives.

2 LITERATURE REVIEW

The properties of proper time interval division are usually given as follows; the time intervals to be selected when extracting a network must be neither small enough to make the network noisy nor large enough to ignore significant time-dependent effects on the network. It is reported that if the time interval is too small, the network may have a lot of noise (Fish & Caceres, 2015; Soundarajan et al., 2016; Sulo et al., 2010). Previously, numerous solutions have been proposed for finding the most appropriate time intervals. Most of those works are dedicated solving this problem for a specific application or a specific data set. Finding proper time intervals for dynamic network extraction has mostly been handled with an empirical procedure. Usually, the authors determine a set of window sizes that correspond to the possible proper time intervals. First, they slide those windows and extract different dynamic networks for each of them. Second, they express each of these networks with a time series generated from its snapshots' features. Third, they analyze those time series, and accordingly, they decide the most robust one. Finally, the most proper time interval corresponds to the window size, which results in the most robust time series. The two most critical points in this approach are (1) to extract a time series with appropriate snapshots' features and (2) analyze these time series with a robust methodology. We overview the state-of-art works with these two perspectives.

The most commonly used features are topological network properties. For instance, in (Sulo et al., 2010), Sulo et al. consider many popular topological properties such as average path length, radius, eccentricity, diameter, density etc. They analyze time series by their variance and compression ratio. They state the proper time interval as the one that balances the minimization of noise and loss of temporal structural information. Thus, they look for the time series whose variance approaches zero. They propose an extended work in (Caceres et al., 2011). In this extended version, the proper time interval is represented as a linear function of network links exhibiting a stationary behavior. Later on, Uddin et al. underline that using network-level

topological properties, e.g. average path length, radius, density etc., can be misleading (Uddin et al., 2017). Although the roles of nodes change, two networks can still have a similar macroscopic view. That is why; they propose to concentrate on a combination of three node-level centrality measures; degree, closeness, and betweenness, to measure the positional dynamicity of nodes. For time series analysis, they use ARIMA and unsupervised learning with the kmeans algorithm. The works mentioned above result in one specific time interval for discretion, where the system is defined. Usually, it might not be the case for many systems. Different time intervals might be needed for a different period of the system. For instance, in (Soundarajan et al., 2016), the authors propose a framework which is called ADAGE. It aggregates the links until the user is given topological property converges. Thus, it creates incrementally consecutive snapshots. Different than most of the works, ADAGE may find different time intervals for different periods. We come across that Fish et al. propose a supervised learning method to attack a similar problem set (Fish & Caceres, 2017). They label critical change points where the system exhibits considerable changes. This approach is successful because it can find different time intervals, but as it requires manual labelling, it is based on subjective criteria and time-intensive.

Monitoring topological properties gives us an idea of the global structural changes in the system. However, one cannot capture the presence or absence of links over time by considering only them. Moreover, most of the topological properties are not normalized. Hence they cannot be evaluated in objective reference points. In the literature, we encounter the similarities of consecutive snapshots are also used as features for extracting time series. Clauset and Eagle (Clauset & Eagle, 2012) propose one of the earliest solutions. They made a time series spectral analysis on not only topological properties as degree and transitivity but also on adjacency correlation coefficient that they developed for measuring nodes' neighborhood similarity between consecutive snapshots. The adjacency correlation coefficient corresponds to the correlation of the adjacency matrix of two consecutive snapshots. Hence they considered node-to-node changes over time. In (K. Darst et al., 2016), the authors underline the roles of similarities of consecutive system events for the detection of time intervals. They look for the peaks in the Jaccard similarity of the system events between a given time and after a precise time interval. Their assumption is very similar to the one that we propose in this work. Nevertheless, they use system events similarity directly and not the consecutive

snapshots' similarity. Krings et al. proposed a very similar solution in (Krings et al., 2012). Different from the approach of (K. Darst et al., 2016), they do not use system events' similarity. They use both topological network properties and Jaccard similarity-based consecutive snapshot similarities. However, when they slide windows, they do not extract separated snapshots, but they aggregate the snapshots. After all sliding process, they extract one big complex network which represents all system activities. At each slide, they examine the results of topological properties and similarity scores. In their works, they concentrate on only link set similarities. However, in this work, we propose node, link, and neighborhood similarities.

Finally, we would like to mention a review, even if it is not directly related to our work. An exhaustive empirical performance evaluation study for network comparison metrics is proposed in (Wills & Meyer, 2019). The authors compare several different metrics based on the adjacency matrix or spectral distance. They have shown that adjacency spectral distance is the most reliable metric for comparing two networks. However, this metric works for networks at an equal size, i.e., two networks have to have the same number of nodes. Two consecutive snapshots do not necessarily have the same number of nodes because the system is dynamic. As time goes by, some new actors can come to the system while some others can leave. That is why; we use three different topological similarity metrics based on the Jaccard similarity of node and link structure and neighborhood of the nodes.

3 DATA SETS

We have listed the most popular data sets for dynamic network extraction and finding proper time intervals in the table 3.1. Here, we only consider social data and skip the biological or other ones. Among them, MIT Reality Mining and Enron are the most commonly used ones. Most of the works explained in the previous section consult these data sets to validate their methodology. In general, the data sets are small-sized. We see that only Belgium Telecom is extensive data with 2.1 million users. For the time spans where the data sets are defined, we come across weekly, monthly, and yearly periods. In our experiments, we use both the WAP log and the Enron email data set. We explain them in the following parts.

Table 3.1: Common Data Sets

Data Set Name	**Size**	**Time Span**
MIT Reality Mining (Sulo et al., 2010; Uddin et al., 2017; Caceres et al., 2011; Clauset & Eagle, 2012; Eagle & , Sandy; Fish & Caceres, 2017; K. Darst et al., 2016)	90 users	9 months
Enron Email (Sulo et al., 2010; Fish & Caceres, 2017; K. Darst et al., 2016)	151 employees	53 months
Haggle Infocomm conference (Caceres et al., 2011; Fish & Caceres, 2017)	41 participants	4 days
UCI Network (Uddin et al., 2017)	not given	not given
Facebook (Soundarajan et al., 2016)	not given	not given
Twitter (K. Darst et al., 2016)	not given	1 week
Belgium Telecom (Krings et al., 2012)	2.1 million customers	6 months
Badge (Fish & Caceres, 2017)	23 employees	1 month

3.1 Wi-Fi Access Point Connections

Data sets such as telecom, email, Facebook, or Twitter are naturally under the social network form. For them, direct relationships between person A and person B can be represented with network links. That is why; these data sets cause relatively less noise or less challenge for dynamic network modelling. However, the WAP log data set does not directly reflect the social relationships but it might help us discover common behavior of people being in the same environment. WAP is studied under its raw form previously for predicting users' position, behavior, or habits (Alessandrini et al., 2017; Rodriguez-Lozano et al., 2016; Zhang et al., 2018). It is also shown that people's physical proximity can be inferred by their connection to the same WAP within a sufficiently small time window (Stopczynski et al., 2014). However, many problems are limiting the handle of raw WAP logs. In (Kjærgaard & Nurmi, 2012), the authors explain in detail the challenges of using WAP logs for extracting social information. Briefly, first, it is noisy data compared to other location data sources such as GPS. There are continuous and random connection breaks because WAP's instantly drop the connections and add them again. Second, the data might contain some misleading information. A person may connect to a powerful WAP, which is far from him. Thus, the position of a WAP is not necessarily reflecting the actual position of the person that connects to. Here, we use it as a case study to find the proper time interval for building the correct dynamic network.

The Wi-Fi tracking data consists of system connection metadata, covering the 2016-2017 fall semester, 137 days in total, that were logged in every 10 minutes by the IT department of Sabancı University. At each logging moment for each access point, device IDs that are connected to that access point are received with time information. Even if logging is made once every 10 minutes, the devices' information is recorded with the exact connection and disconnection times in the meantime when the logs are received. Before explaining the details of data, we want to give details about the campus life in this university because the connections directly reflect those real-life phenomena. The campus consists of 3 major areas: dormitories, faculty buildings, and utility centers. The lessons start at 8:40 and end at 19:30. Each lesson is 50 minutes, and there is a 10 minutes break between them.

The record, 9 million in total, includes *device ID, connection/disconnection*

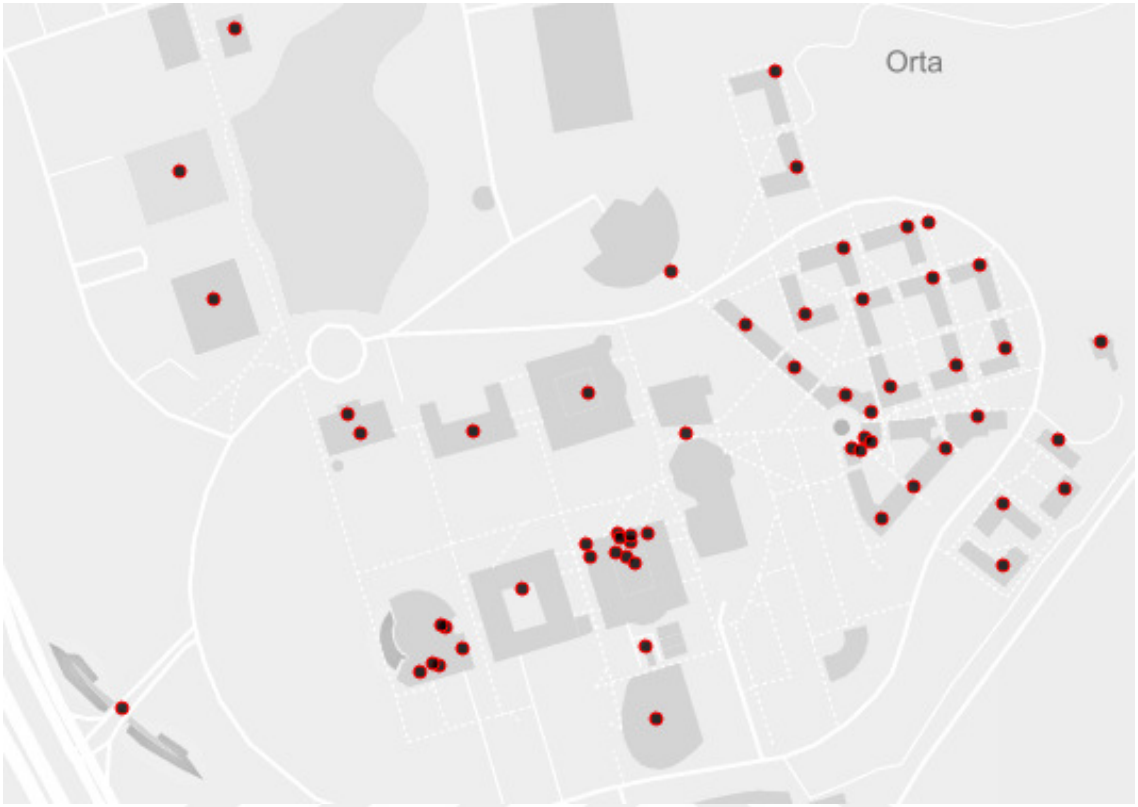


Figure 3.1: Spatial Wi-Fi access point distribution in the campus.

timestamps and *WAP name*. The device IDs, 68114 in total, were anonymized by assigning unique values to each MAC address that connects to any WAPs on the campus. We create a node for each device ID that appears in the system. If two devices connect to the same Access Point for a given time interval, we put a link between them. A link between two nodes might be a sign that those devices are at the same place. The majority of the WAPs are located in dormitories and faculty buildings, while utility centers possess a small portion. The distribution of 613 WAPs can be observed in figure 3.1. Due to condensed building settlements, on average, a device is connected to 17.25 WAPs with a duration of 39.19 minutes. A small sample of five records is given in table 3.2.

Collected data from WAPs have much noise. Before we extract networks, we need to clean it. Here, we work on two main issues. First, we remove the records whose connection and disconnection timestamps are not represented. Then, we merge the records representing the continuity of the connection but dropped because of WAPs' properties. The records #1, 2 and 3 in table 3.2 are example of such continuity. If a device is dropped, it can connect to either the same WAP (as #1 and 2) or another

Table 3.2: Example of Sabancı Wi-Fi Access Point Data

Device ID	Access Point	Connection	Disconnection
10083	YurtA6Kat3A	20161204 12:31:00	20161204 21:59:00
10083	YurtA6Kat3A	20161204 21:59:00	20161204 22:43:00
10083	YurtA6Kat3C	20161204 22:43:00	20161204 22:50:00
10084	FASSG0533c	20161208 23:51:00	20161209 00:42:00
10457	YurtD1B2237	20161205 11:10:00	20161205 15:25:00

(as #2 and 3) whose location is close to the previous WAP. We consider the first case and merge these rows in the data. For those records, the following procedure is performed.

- For each device, we order the data according to their connection time,
- If two consecutive records have the same WAP names and the previous records' disconnection time is equal to the next records' connection time, we merge these two records.
- We update the record list and continue till all records are scanned.

3.2 Enron Emails

Enron is a well-known dataset that has been widely studied in network science and text mining. It has been used frequently in the literature in the form of different static and dynamic networks. This study generates dynamic networks by dividing raw Enron data into different time intervals to find the most proper time interval for its modelling. Previously, Enron data was used by Clauset and Eagle for a similar purpose (Clauset & Eagle, 2012).

Raw data consists of the emails of 151 Enron company employees. These emails are sent between 1997 and 2002. When we generate networks, we consider not only company employees, but also all mail addresses at every From and To fields in the corresponding emails as network nodes. When all the from and to fields are used for

all emails between 1997 and 2002, 28802 nodes are generated. A link is established between the nodes which appear at From and To fields for a given time period. Mailing networks are usually represented by directed networks. However, our aim in this study is to determine the proper time intervals. Therefore, snapshots are undirected.

4 PROPER DYNAMIC NETWORK EXTRACTION PROBLEM

In this section, we first introduce the preliminaries, then, we propose a problem definition for the proper extraction of dynamic networks for any system.

4.1 Preliminaries

Let us consider $\mathcal{C}$ is a complex system which includes interacting objects. We note that the beginning and end timestamps of the interactions are t_1 and t_θ respectively. The interactions change in a *continuous time interval* $[t_1, t_\theta]$. We model $\mathcal{C}$ under the form of a complex network $\mathcal{G}$. If we tackle all the interactions occurring in $[t_1, t_\theta]$ at once, we extract only one network which is called a *static network*. In this case $\mathcal{G} = (\mathcal{V}, \mathcal{L})$ is a pair such that $\mathcal{V}$ is its set of nodes, including the representation of all the objects appearing in the system between t_1 and t_θ, and $\mathcal{L} \subseteq \mathcal{V} \times \mathcal{V}$ is its set of undirected links, representing the interactions between objects. In fact, a richer complex network modelling is also possible for temporally changing systems. We call a *dynamic network*, $\mathcal{G} = \langle G_1, \ldots, G_\theta \rangle$, as a finite sequence of chronologically ordered static networks G_i $(1 \leq i \leq \theta)$. We call each of these static networks, G_i, specifically snapshots. More formally, a *snapshot* $G_i = (V_i, L_i)$ is a pair of node set and link set where i is a sub-interval in $[t_1, t_\theta]$. Hence, V_i is the set of nodes representing the objects appearing during i in $\mathcal{C}$ and $L_i \subseteq V_i \times V_i$ is the set of their links. *Duration* of the snapshot G_i is the length of sub-interval i.

To model a complex system under the form of a dynamic network, one needs to segment continuous time interval into a chronologically ordered series of sub-intervals. A time interval can be divided into *discrete* sub-intervals as well as *overlapping* ones. Moreover, dynamic networks can be extracted by choosing different duration for different snapshots. Considering all these options, there could be infinite numbers of different *candidate dynamic networks*, which all are extracted

for the same system. Thus, finding the most appropriate dynamic network extraction among all candidate ones demand a search operation in an infinite set. That is why; we do not search the best solution, but we look for a compact and informative dynamic network. Such a network has few numbers of snapshots reflecting all critical information about the system. We accept that when the system is stable, i.e., its components do not change, there is no new information, so, no need to extract a new snapshot. Moreover, when the stability is broken, i.e., a considerable change is observed in system components, a new snapshot should be extracted. This problem can also be seen as finding proper sub-interval division of continuous-time interval where the system is considerably changing. One such division has been found, afterwards, one can extract related snapshot. In this work, we look for a discrete sub-internal division. Let us focus on this problem and define it in the next subsection.

4.2 Problem Definition

Let us define a measure S, which quantifies the *stability* of a complex system as the number of its unchanging components, i.e., objects and interactions, at given sub-interval. If S is high, then $\mathcal{C}$ is stable from the beginning till the end of the given sub-interval.

The *shortest stable duration*, ϵ, of $\mathcal{C}$ is the shortest duration in which the system stays stable. We can accept that its value is too low for any complex system. It corresponds to a moment where there are very few changes in the system objects and interactions from the beginning and end of that moment.

In order to determine *the longest stable duration*, $\triangle_i$, for given beginning time t_i, we add ϵ to t_i until the stability of system breaks. Thus, $\triangle_i$ is a multiple of ϵ. Considering that ϵ corresponds to a concise moment, the completeness of $\triangle_i$ is the longest duration is evident. Once the system loses its stability, the rest part could be seen as a new system in order to determine the new longest stable duration, $\triangle_{i+1}$, of the beginning time $t_i + \triangle_i$. Hence, $\triangle_{i+1}$ could be found as another multiple of ϵ where the system's stability is broken.

By this iterative approach, we define *proper duration sequence*, $\triangle$ =

$\langle \triangle_1, \ldots, \triangle_{\theta-1} \rangle$ is the sequence of the longest stable duration in which $\mathcal{C}$ is stable. Accordingly, $\langle [t_1, t_1 + \triangle_1], \ldots,](t_1 + \sum_{i=1}^{\theta-1} \triangle_i), (t_1 + \sum_{i=1}^{\theta} \triangle_i)] \rangle$ is *proper discrete sub-interval division* of $[t_1, t_\theta]$. $\mathcal{C}$ is stable at its longest duration inside the period of any member of this sequence, however between consecutive members, $\mathcal{C}$ is unstable. In this way, we define a *proper dynamic network* of $\mathcal{C}$ as $\mathcal{G}_\triangle = \langle G_{[t_1, t_1+\triangle_1]}, \ldots, G_{]t_1+\sum_{i=1}^{\theta-1} \triangle_i, t_1+\sum_{i=1}^{\theta} \triangle_i]} \rangle$. Each member of $\mathcal{G}_\triangle$ is the snapshot for given discrete sub-interval. By this definition, we expect that for any snapshot, $\mathcal{C}$ stays stable at its longest duration till the beginning of next snapshot. Hence *proper dynamic network extraction* problem is the problem of finding proper discrete sub-interval division of the time span where the system is defined. Here we propose a continuous time discretization considering the stability of system. In the beginning of this section, we mention about any measure S for determining if the system is stable. Let us deepen this concept in following section because it constitutes the major decision criteria for dynamic network extraction.

5 MEASURING THE STABILITY

In the previous section, we introduce S as any stability metric, which calculates the number of unchanging components of $\mathcal{C}$. In order to extract a proper dynamic network from $\mathcal{C}$, one needs to shift a window of ϵ length starting from t_1 in the direction of time flow. At first shift, we calculate S, from t_1 till $t_1 + \epsilon$. If the S is showing that the system stays stable, we shift the window again. We again calculate S from $t_1 + \epsilon$ till $t_1 + 2\epsilon$. This process continues until S is showing that stability is broken. Accordingly, S should have some properties for being used for proper dynamic network extraction. We list those properties in table 5.1.

Table 5.1: Properties of Stability Measure S

Property 1	S must have two parameters; previous time point, t_{prev}, and following time point, t_{next}
Property 2	S must quantify the amount of components that stay same at t_{prev} and t_{next}
Property 3	There should be a bound of S to decide the stability of the system
Property 4	The choice of shortest stable duration, ϵ, i.e. the duration between t_{prev} and t_{next}, has a critical effect on S

According to *Property 2* defined in table 5.1, S relates to the components of $\mathcal{C}$. Those components consist of system objects or their interactions. Hence, S can also measure the stability of their abstraction, nodes, and links, respectively. Considering *Property 1, 2, and 3* together, S can be any network similarity measure whose objective bound is defined. From network perspective, S measures the similarity of snapshot at t_{prev} and t_{next}. In the literature, we see that the similarity of complex networks was discussed in (Schieber et al., 2017; Shimada et al., 2016; Wills & Meyer, 2019). They focused on the spectral properties of networks. However, comparing

two networks with their topology gives us only a general idea about how similar these networks are. To ensure *Property 2*, we want to focus on each unique node and link. Thus, we need to compare two node sets and link sets rather than an overview of network topology. We propose to use *Jaccard Similarity* of node sets and link sets. Jaccard Similarity is a very well-known and commonly used metric for quantifying the number of common parts of two different sets (Jaccard, 1912). There are many different variants of it. The one we use here is the measure of the proportion of the amount of elements that both appear at two sets to the amount of elements that are seen at least at one set. We use Jaccard Similarity on three different types of sets. For the sake of comprehensibility, we use different similarity names dedicated to different types of sets.

5.1 Similarity Metrics

The first similarity is the *Node Similarity*, $S_{node}(t\_prev, t\_next)$. It is Jaccard Similarity of nodes sets of previous and next snapshots. The equation is given in equation 5.1.

$$S_{node}(t\_prev, t\_next) = \frac{\mid V_{t\_prev} \bigcap V_{t\_next} \mid}{\mid V_{t\_prev} \bigcup V_{t\_next} \mid} \tag{5.1}$$

The second similarity is the *Link Similarity*, $S_{link}(t\_prev, t\_next)$. It is Jaccard Similarity of links sets of previous and next snapshots. Its equation is given in equation 5.2.

$$S_{link}(t\_prev, t\_next) = \frac{\mid L_{t\_prev} \bigcap L_{t\_next} \mid}{\mid L_{t\_prev} \bigcup L_{t\_next} \mid} \tag{5.2}$$

Let $N_{t\_prev}(v)$ and $N_{t\_next}(v)$ are the first order neighborhood of $v \in V_{t\_prev} \bigcap V_{t\_next}$ at previous and next snapshots respectively. We define for a given node, v, its *neighbor stability*, $\delta(v, t\_prev, t\_next)$ as Jaccard Similarity of $N_{t\_prev}(v)$ and $N_{t\_next}(v)$. The equation is given in equation 5.3.

$$\delta(v, t\_prev, t\_next)\frac{\mid N_{t\_prev}(v) \bigcap N_{t\_next}(v) \mid}{\mid N_{t\_prev}(v) \bigcup N_{t\_next}(v) \mid} \tag{5.3}$$

The third similarity is the *Neighborhood Similarity*, $S_{neighbor}(t\_prev, t\_next)$. It is the average *neighbor stability* of the common nodes in previous and next snapshots. Its equation is given in equation 5.4. $S_{neighbor}$ uses nodes as units. It reflects the average neighborhood stability over nodes while S_{link} concentrates directly on links. When network node numbers do not change very much and network centralization is not very high, we encounter that S_{link} and $S_{neighbor}$ contains similar information. If network node structure change is considerable, the result scores of those two metrics fall apart. For example, for a star shaped network whose most of its non-central nodes leave as time goes by, its S_{link} and $S_{neighbor}$ would be very different.

$$S_{neighbor}(t\_prev, t\_next) = \frac{1}{\mid V_{t\_prev} \bigcap V_{t\_next} \mid} \sum_{v} \delta(v, t\_prev, t\_next) \tag{5.4}$$

Up to now, we defined the problem of extracting a proper dynamic network for a complex system as finding proper discrete sub-interval division of the time span where it is defined. We proposed to use a similarity metric and gave its properties in table 5.1. We proposed to use Jaccard Similarity, which already ensures *Property 1 and 2* by its definition. For the rest of this section, we will focus on *Property 3 and 4* for Jaccard Similarity compatibility.

5.2 Objective Limit of Similarity Metrics

The objective limit corresponds to the point that the system begins to change considerably. Accordingly, we decide if the extraction of a new snapshot is necessary or not. For higher S than its limit, the system stays stable, i.e., no need to extract new snapshots, while for lower values, the system change considerably. So, a new snapshot can be extracted. In our problem, we look for lower bound of Jaccard Similarity for it ensures *Property 3* in table 5.1. Jaccard Similarity is a normalized metric which is defined between 0.0 and 1.0. If it is 0.0, there is not any common elements between compared sets. In other terms, two sets are entirely different. If

it is 1.0, then two sets include completely the same elements. However, which value of Jaccard Similarity signifies enough similarity of two sets?

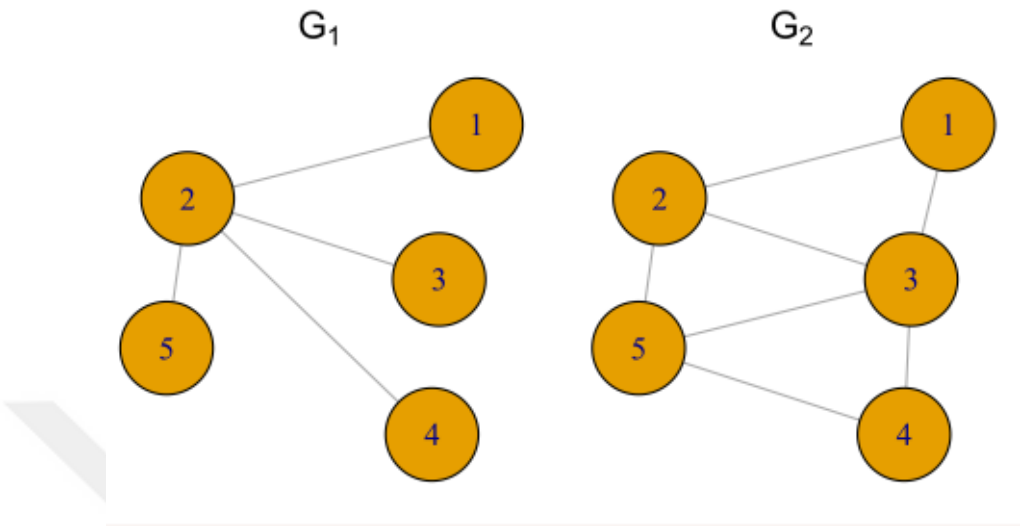


Figure 5.1: Two consecutive simple snapshots.

For instance, for two consecutive snapshots given in figure 5.1, similarity scores are $S_{node} = 1.0$, $S_{link} = 0.375$ and $S_{neighbor} = 0.366$. The changes over the link structure affect S_{link} and $S_{neighbor}$ directly. Moreover, the quantity of S_{link} and $S_{neighbor}$ is not the same. Nevertheless, how can we decide if these two consecutive snapshots are similar or not? What are the bounds of being similar or being different? Let us now concentrate on the statistically objective limits of Jaccard Similarity to decide the system's stability.

In (Real & Vargas, 1996), the authors study the limits of Jaccard Similarity with the probabilistic approach for biological similarity of species for the taxonomy. They propose a null model that quantifies how much random the two studied sets are in terms of their common element numbers. They also give the list of critical values of Jaccard Similarity in terms of total number of elements for small sized sets in (Real, 1999). They work with the samples of species. When two species are compared, the total possible number of attributes which can be observed is noted as N. A and B are the numbers of attributes present in samples of first and second species respectively and C is the number of attributes present in both species in the samples. The formula of the null model is given in equation 5.5.

$$P = \frac{\sum_{x=0}^{C} \binom{A+B-x}{x}}{\sum_{x=0}^{min(A,B)} \binom{A+B-x}{x}} \tag{5.5}$$

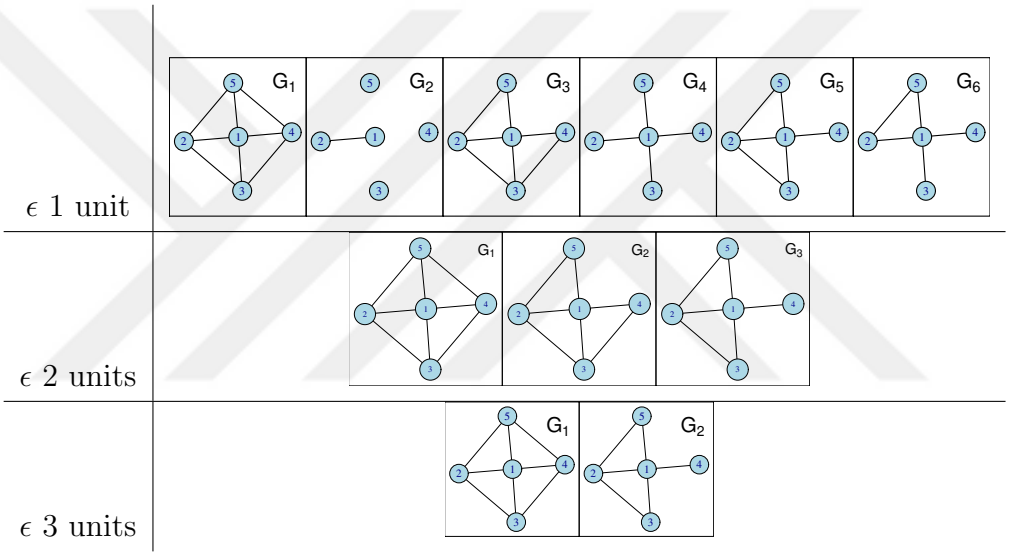
We can interpret A and B as the number of elements in two sets. The common elements may range from 0 to the minimum value A and B. Thus, the denominator of equation 5.5 represents the number of all possible intersection states that two sets can have. Similarly, the nominator expresses the number of all possible intersection states that two sets can have when knowing that two sets have at most C elements in common. Finally, equation 5.5 gives the limit of having C elements in common by chance. If two sets are similar, their Jaccard Similarity should exceed the limit of being by chance. In our problem, we will compare two consecutive snapshots' node sets or link sets. Because node numbers or link numbers are larger than the table given in (Real, 1999), we apply the formula given in equation 5.5. In our case, A and B correspond to the number of nodes or links in consecutive snapshots, and C corresponds to the number of common elements in node sets or link sets for consecutive snapshots. For now, we do not consider $S_{neighbor}$ because $S_{neighbor}$ is an average score calculated for every node. The formula given in equation 5.5 is suitable for computing a limit for δ for every node, but one cannot use it for $S_{neighbor}$.

5.3 Setup for Empirical Validation of Similarity Metrics

The choice of ϵ has a critical effect on determining a system's stability. It can be seen as a temporal aggregation unit for accumulating nodes and links. For instance, if the system includes data for one day and ϵ is chosen as one day, we extract one snapshot, including all system objects and their interactions. Thus, we miss out on the dynamic structure of the system. Or, if ϵ is chosen as one millisecond, we extract over 5 million snapshots that each consecutive one are very similar (if there is not a critical point in milliseconds). In this case, the accumulation of ϵ might not be significant, and the system might be seen as stable when analyzing. But it is just an artifact of the wrong choice of ϵ. In table 5.2, we show how the choice of ϵ can change the resulting network structure on an example. We listed three dynamic networks for the same system. The first one is generated for the shortest ϵ, i.e., when ϵ is one unit. The network has six snapshots. Consecutive ones seem dissimilar with

each other. In this representation, the network is very dynamically changing. If we increase ϵ to two units, we have three snapshots. They look very much like each other. The dynamic change of previous representation does not seem any more for this one. When ϵ is chosen as three units, the network has two snapshots. They look like each other. We cannot catch the dynamic changes at this one as well. In this example, network dynamism is affected negatively when we increase ϵ. In another example, if the system changes very slowly or if it stays stable for a too long time, high ϵ might bring advantages.

Table 5.2: An Example of consecutive snapshots extracted for different ϵ



Different values of ϵ have a critical effect on measuring systems' stability. We perform an experimental analysis to show how the metrics we propose behave according to different ϵ. Our main purpose in the experiments is to show the usability of the proposed metrics in measuring system stability. We also consider *adjacency correlation* metric, γ, proposed in (Clauset & Eagle, 2012). It is also developed for measuring the similarity of consecutive snapshots. Adjacency correlation is the correlation of adjacency matrices of consecutive snapshots. The quantification of correlation between the row elements that are not zero at least once in either of the adjacency matrices. We use γ as baseline and make a comparative performance analysis between γ and proposed similarities. In the experiments, we follow a simple procedure. First, we extract dynamic networks from data described in section 3 by using different ϵ. Second, we examine the properties of the extracted networks. Third, we calculate similarity metrics by considering their significance and also γ.

Finally, we empirically analyze the results and interpret them. Because the WAP log is a new data set that has not been analyzed before, we examine its topological properties in detail for different ϵ. We study ten network topological properties which are commonly used in network analysis listed in table 5.3 with their general formulas[1][2][3][4]. We observe the evolution of their values taken for different snapshots as time goes by under the form of time series. For the rest of this article, we call them signals.

Table 5.3: Studied topological properties and their formula

Name	Formula
Node Number	$n = \|V_i\|$
Link Number	$m = \|L_i\|$
Network Density	$D = m/(n(n-1)/2)$
Average Degree	$\langle k \rangle = 2m/n$
Number of Connected Components	$c = \|c_i\|$ c_i is the set of maximal connected sub graphs of G_i
Average Path Length	$z = 1/n \sum_{u,v \in V_i} d(u,v)$, $d(u,v)$ is the distance (shortest path) between nodes u and v
Diameter	$diam = max_{u,v \in V_i}(d(u,v))$
Transitivity	$CC = 1/n \sum_{u \in V_i} triangles(u)/triples(u)$, $triangles(u)$ is the number of triangles connections where u is linked $triples(u)$ is the number of connected triples(including triangles)
Closeness Centrality	$C_c = 1/\sum_{u \neq v} d(u,v)$
Betweenness Centrality	$C_b = \sum_{u \neq v \neq w} g_{vw}(u)/g_{vw}$ $g_{vw}(u)/g_{vw}$ is the probability of picking shortest path between v and w which contains u (0 if $g_{vw} = 0$)

For the WAP log dataset, we analyze 137 days of data with regular and irregular days. Here, we evaluate only seven regular days (from December 4 to 12, 2016) for simplicity. These are the dates where similarity scores were relatively high

[1]Newman, M. E. J., The structure and function of complex networks, 2003

[2]Barabasi, A., Albert, R., Statistical mechanics of complex networks, 2002

[3]Freeman, L. C., Centrality in Social Networks I: Conceptual Clarification, 1979

[4]Dorogovtsev S.N., Mendes J.F.F., Evolution of networks, 2002

compared to other days. We use eight different time intervals in minutes, $\epsilon = \{1, 3, 5, 10, 30, 40, 60, 1440\}$. We choose these values to respect the minimum time interval seen in data (1 min.). We also consider the most frequently seen time intervals in the data (3 and 5 min.), break times between lectures (10 min.), proper walking duration in the campus (30 mins.), proper lecture duration (40 and 60 min.) and one day (1440 min.). For Enron, we analyze 1178 days of data. Snapshots for short ϵ like a period of minutes or hours were too sparse and they have non realistic topological properties. That is why; we use a longer period, such as daily scales for time intervals. We choose $\epsilon = \{1, 7, 15, 30, 90\}$ days. We respect to the daily, weekly, monthly and quarterly periods for companies. We compute S_{node}, S_{link} and $S_{neighbor}$ for all consecutive snapshots for each member of the experimental set. We also compute S_{node_limit} and S_{link_limit}. Each metric's results is a real-valued time series vector whose length is equal to the number of the related snapshot - 1. We interpret these time series from their graphical.

6 RESULTS

We first interpret the results for the WAP data set, and then we observe Enron results. In the following sections, the readers will find the results of topological properties, similarity metrics, critical limits of similarity metrics for WAP snapshots, and similarity metrics with their critical limits for Enron respectively.

6.1 Topological Properties of WAP Snapshots

We show average values of topological properties for different ϵ in figure 6.1. From the network science perspective, the results except for the one for 1440 are realistic. Snapshots have similar topological properties scores with well-known social and information networks. As ϵ increases, on average, we have more crowded, denser, and more connected snapshots with a shorter distance between its nodes. The most striking point is that when ϵ is 1440, the snapshots have, on average, quite a different topology than those produced for other ϵ. As the average density is too high (~ 0.15) with a very high average degree (892) and relatively lower transitivity (~ 0.59), we state that snapshots do not exhibit realistic behavior. It seems this ϵ is too high for this system. Thus, we do not examine their results for further analysis. We show the change of each topological properties in function of ϵ in related figures (see figures 6.2, 6.3 and 6.4). A red line on each plot shows the average value of the related signal. Accordingly, for almost all topological properties, we obtain stationary signals with the daily cycle on weekdays. The signal routine seems to be broken at weekends, corresponding between 10 and 12 December. For these days, topological properties' values are lower than the general range. Moreover, signal trends are different from other days.

As recent studies reveal, circadian rhythms of human activity from electronic records (Aledavood et al., 2015; Song et al., 2010; Jo et al., 2012), our findings are also

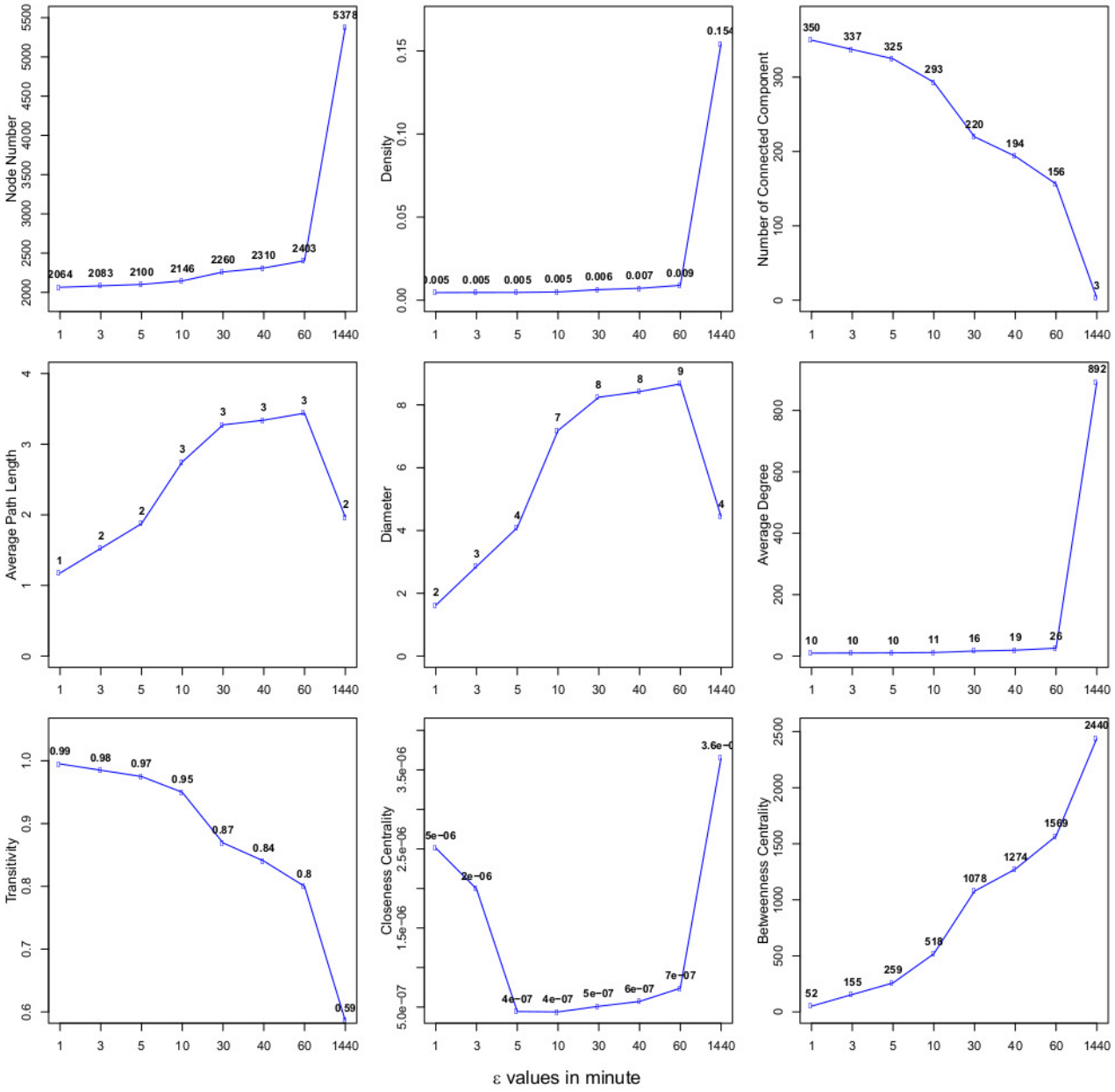


Figure 6.1: Average values of topological properties. The x-axis represents the corresponding ϵ and the y-axis represents the average value of the corresponding topological property.

supporting them partially. In (Aledavood et al., 2015), the authors underline two major periods of circadian rhythm; *day time* and *night*. They did an exhaustive experiment by considering mobile phone activity and also questionnaires. They reveal that different social profiles exhibit different behavioral patterns at those periods. Some people are active at night while some others are not. In our results, signals exhibit a daily cycle in campus life too. Considering all topological properties,

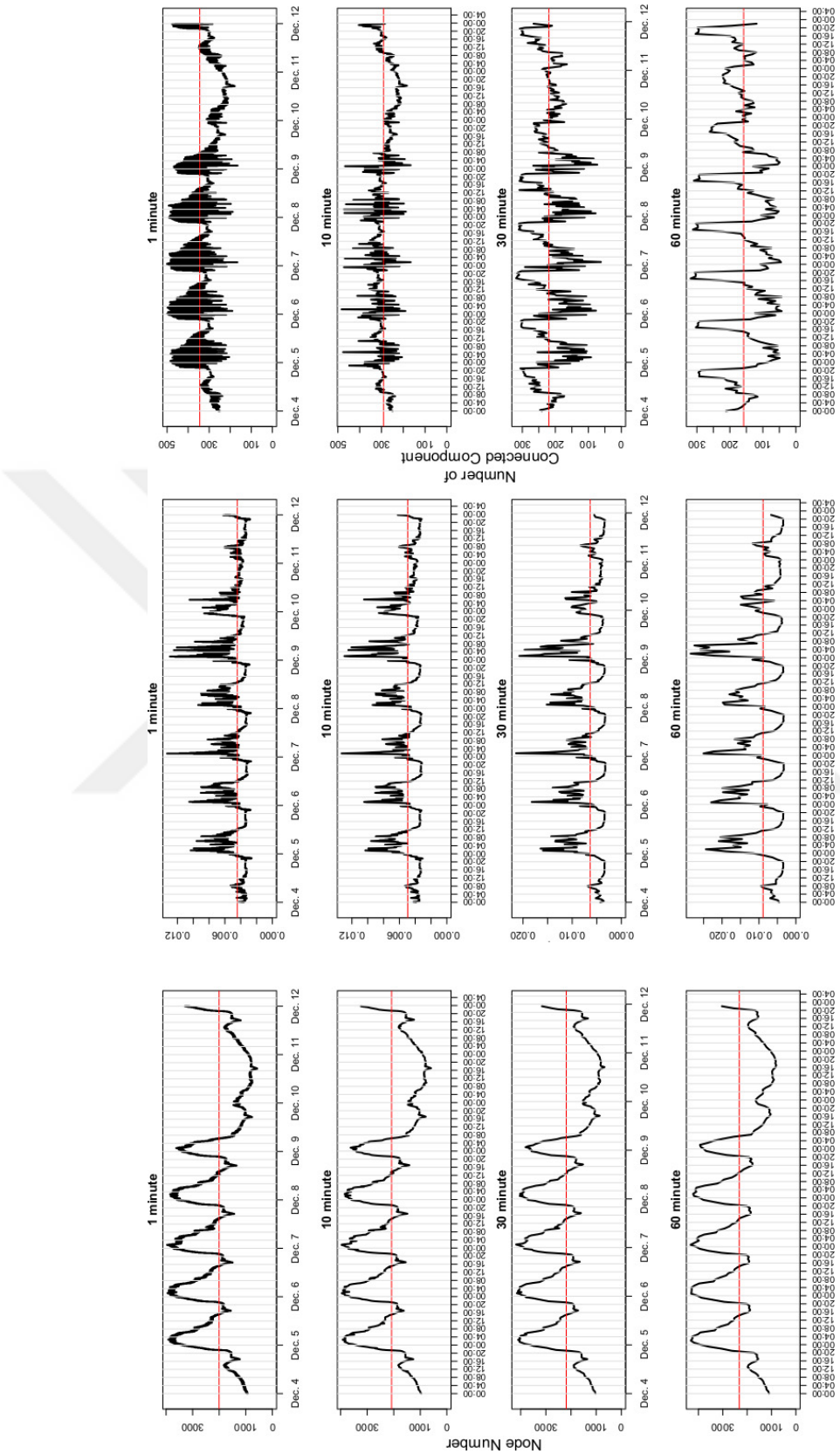


Figure 6.2: Signals for the topological properties of the different dynamic networks extracted for WAP logs in a week between 4-12 December 2016. Each plot shows the results for different ϵ. The X-axis shows both date and time information. Y-axis shows the results of node number, density, and number of connected components for the left, middle and right blocks, respectively.

in general three critical time intervals that the cycle occurs are (1) between [00 : 00 − 02 : 00] and [08 : 00 − 10 : 00], (2) between [08 : 00] and [17 : 00 − 20 : 00] and (3) between [19 : 00 − 21 : 00] and [00 : 00 − 02 : 00]. Those time intervals correspond to three major periods of a day; *night*, *day time* and *evening to night* respectively. Ascending and descending parts of the signals correspond to the hours when the campus gets active and passive.

Looking at details inside these circadian rhythms of campus, we distinguish different social groups appearing on the campus. For instance, at *night* period, campus seems very active. People are getting into campus or walk around, connecting to different WAP's. Most of the night activities that increase and decrease at the topological properties are due to WAP connections in the dormitory area. In the *day time*, the campus is again active and crowded, but this time, not only in the dormitories but also at faculty buildings and utility centers. At *evening to night* period, the campus seems very calm with very few activities. This period corresponds to the hours when the lessons have finished, employees and faculty members leave the campus, and students go back to the dormitories. We also remind that weekend activity is very low. Interpreting all these findings, we distinguish two major groups in the campus; *people staying in the campus* and *people staying off campus*. Most of the people from the first group leave the campus at the weekend, which results in very low activity on the campus.

Evaluating the results in the context of ϵ effect, lower values than 30 minutes causes the generation of too many snapshots. As a result, topological properties' signals become noisy. The higher the ϵ, the clearer the signal is. However, when ϵ is too high (e.g. 1440 minutes), there are very few snapshots resulting in minimal information for discovering campus life's circadian rhythms. A value between those two limits can be a proper time interval for snapshot extraction. In the following section, we examine the results of proposed similarity metrics with Clauset's adjacency correlation coefficient.

6.2 Similarity Results of WAP Snapshots

We represent the average scores of similarity metrics and adjacency correlation coefficient, which are calculated for each ϵ in figure 6.5. Accordingly, similarity

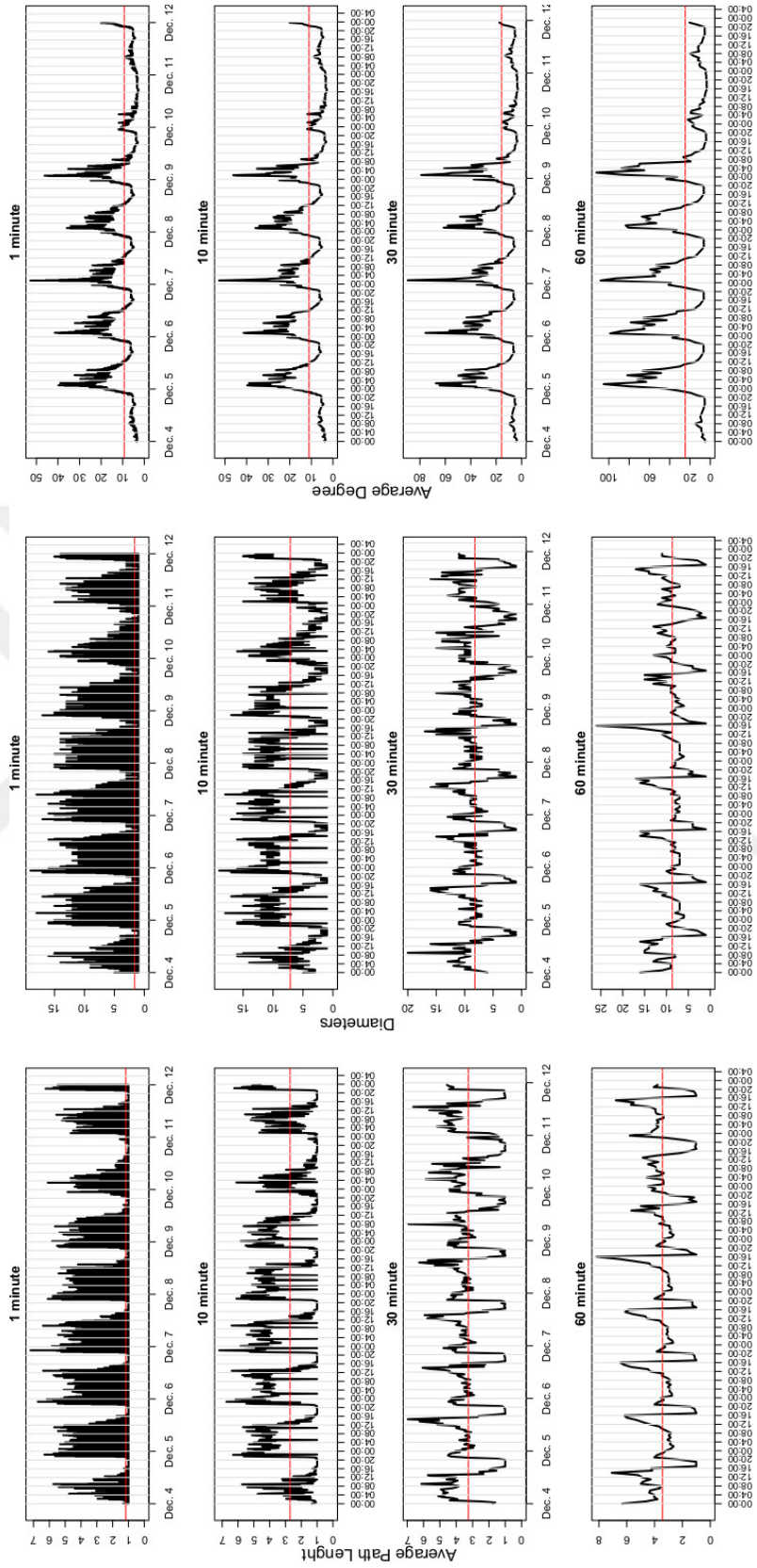


Figure 6.3: Signals for the topological properties of the different dynamic networks extracted for WAP logs in a week between 4-12 December 2016. Each plot shows the results for different ϵ. The X-axis shows both date and time information. Y-axis shows the results of average path length, diameter, and average degree for the left, middle, and right blocks, respectively.

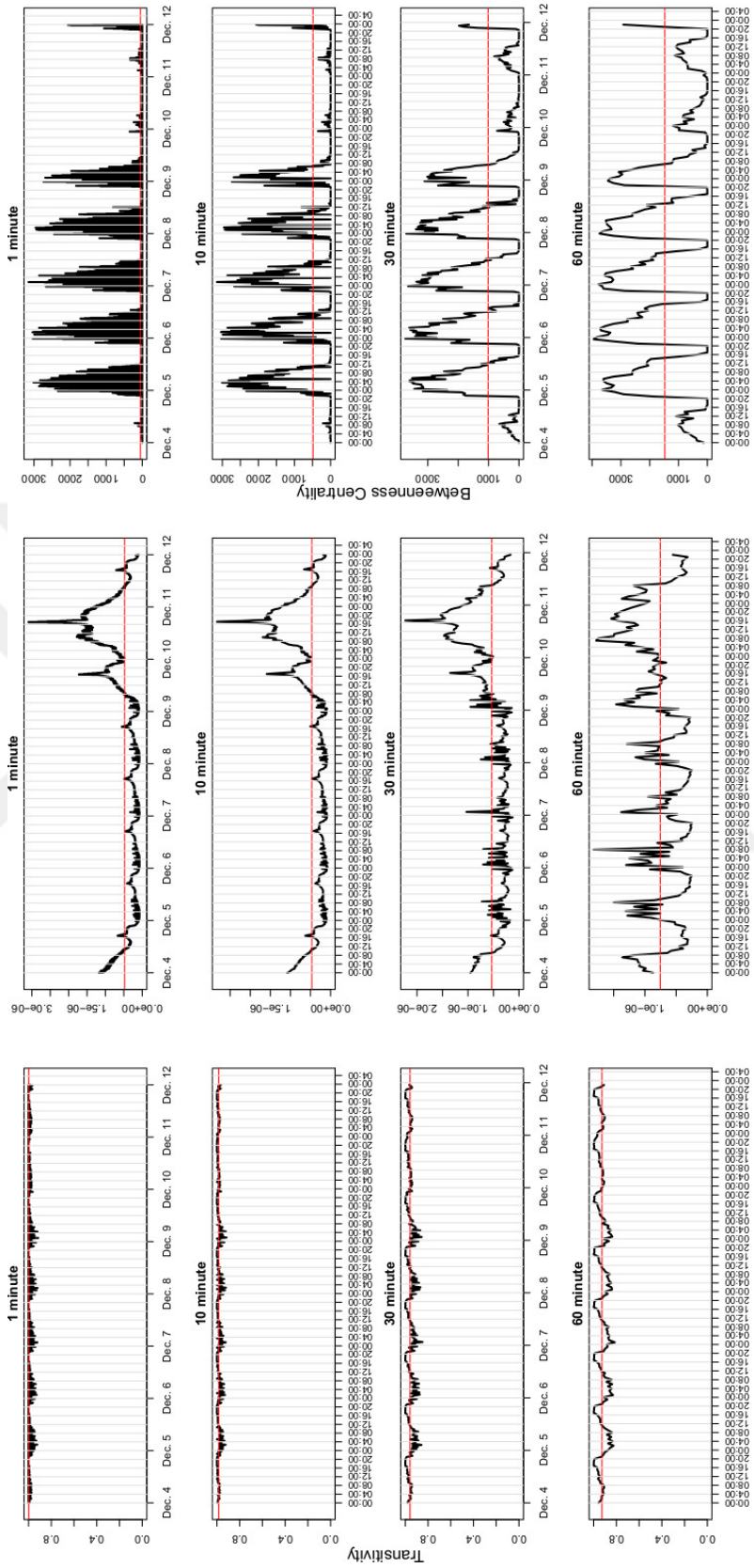


Figure 6.4: Signals for the topological properties of the different dynamic networks extracted for WAP logs in a week between 4-12 December 2016. Each plot shows the results for different ϵ. The X-axis shows both date and time information. Y-axis shows the results of transitivity, closeness centrality, and betweenness centrality for the left, middle, and right blocks, respectively.

scores' reaction to ϵ increase looks like each other, while adjacency correlation behaves differently. We observe all similarity scores decrease non linearly as ϵ increases from 1 to 60 minutes. However, the adjacency correlation first increases when ϵ increases until 10 minutes. Then, it stays stable as ϵ increases. This means that the adjacency correlation coefficient, on average, cannot distinguish the effect of the selected time interval on the similarity of the snapshots for $\epsilon > 10$. Regarding similarity metrics, it seems the most and least sensitive ones are S_{link} and S_{node}, respectively. We observe the highest and the lowest decreases respectively on them. For $\epsilon > 30$, S_{link} and $S_{neighbor}$ gets lower than 0.80. Especially, when ϵ is 1 hour, snapshots seem to have very different link structure ($S_{link} \sim 0.36$) while most of the network nodes stay same ($S_{node} \sim 0.84$).

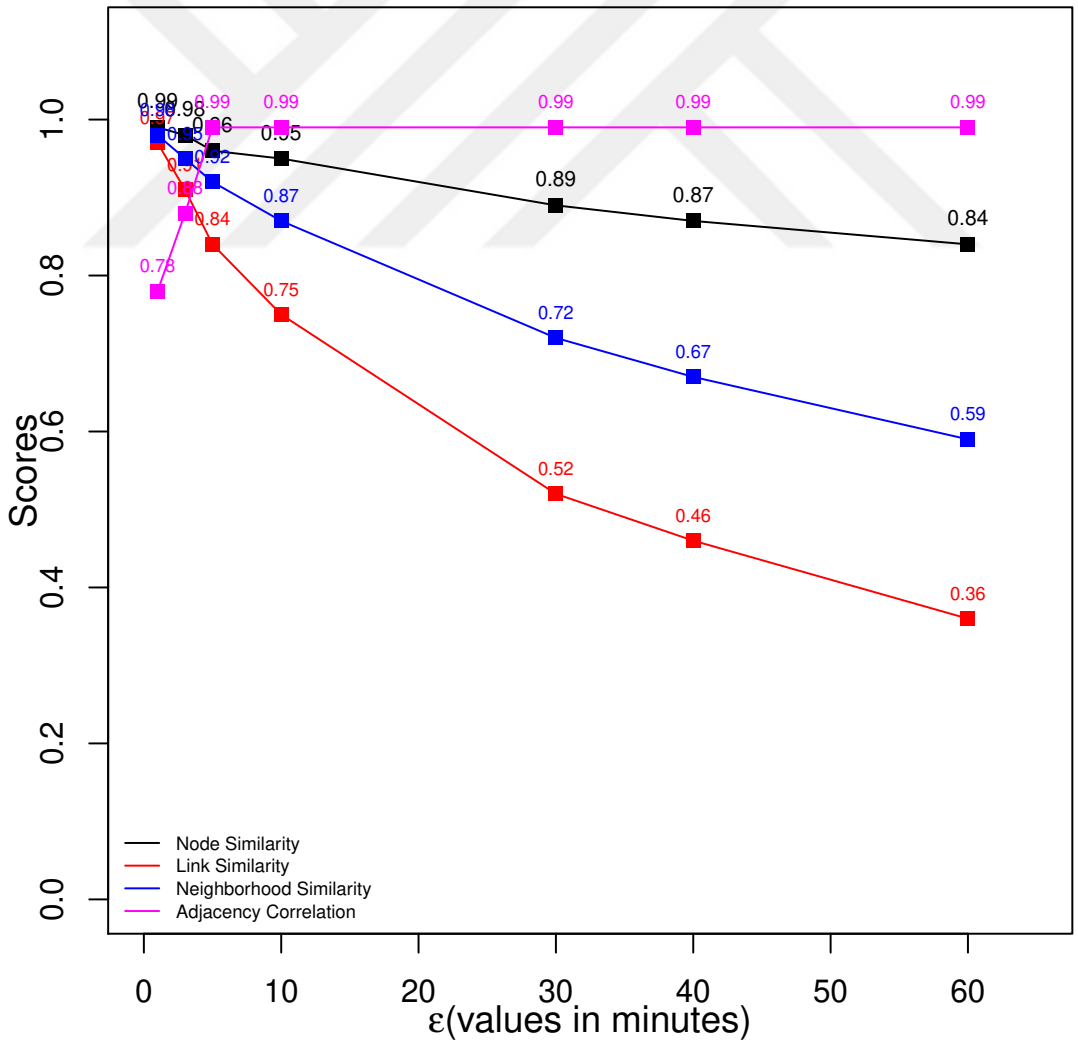


Figure 6.5: Changes on average similarity scores and γ with respect to ϵ.

Focusing on each metrics' signal (see figure 6.6) in detail, we notice that, like topological properties, similarity metrics' signals are also stationary and have cycles. However, the details of γ signal seem different from them. As we noticed on their average values, it does not seem to be sensitive to ϵ. When ϵ is lower than 10 minutes, its signal seems very noisy, while it is not informative for higher values. It takes the same values as time goes by and does not reflect the campus's circadian rhythm at its signals. However, similarity metrics results support the circadian rhythm that is explained by interpreting topological properties. Accordingly, the activity in the campus *increases* from 07 : 00 till 18 : 00, *decreases* from 18 : 00 till 00 : 00 and *stays stable* from 00 : 00 till 07 : 00. These time intervals are compatible with different parts of the day we find when examining the topological properties. Therefore, when we look at the topological properties, it gave us more vague intervals. Similarity metrics reveal clearer time intervals.

We represent the effect of ϵ on S_{link} with a heat map in figure 6.7. Since the metric where we see the effect of ϵ most clearly is S_{link}, we interpret it through its heat map. Nevertheless, the results are valid for all proposed metrics. The heat map is prepared based on 60 per minute period. If the ϵ examined is less than 60 minutes, the result is colored by taking the average S_{link} results for each 60 minutes. As expected, the higher the ϵ, the lower the S_{link}. However, it also starts to discriminate different periods of the day more significantly. S_{link} shows larger variations over a wider range than other metrics. In general, ϵ lower than 10 minutes causes noisy signals for all similarities. These time intervals are too short to distinguish the general differences of snapshots. Short periods can be good intervals to follow up on any specific event. However, to understand public life on the campus, they generate noisy results. 60 minutes look like a good duration for understanding the system's daily routine and clearness of the signal. $S_{neighbor}$ on all ϵ is higher at the weekend than the weekdays. We catch this detail at S_{link} only when ϵ is 5 or 10. While examining the topological properties, we mentioned that the campus is less active on the weekend. Now, $S_{neighbor}$ and S_{link} being higher on weekends also reveals that the campus is more stable and calmer these days. We distinguish the range of the signals at those days are narrower. Moreover, the similarities are higher than other days, especially when ϵ is between 10 and 60. This might be because fewer people stay on the campus, and they mostly stay in the dormitories.

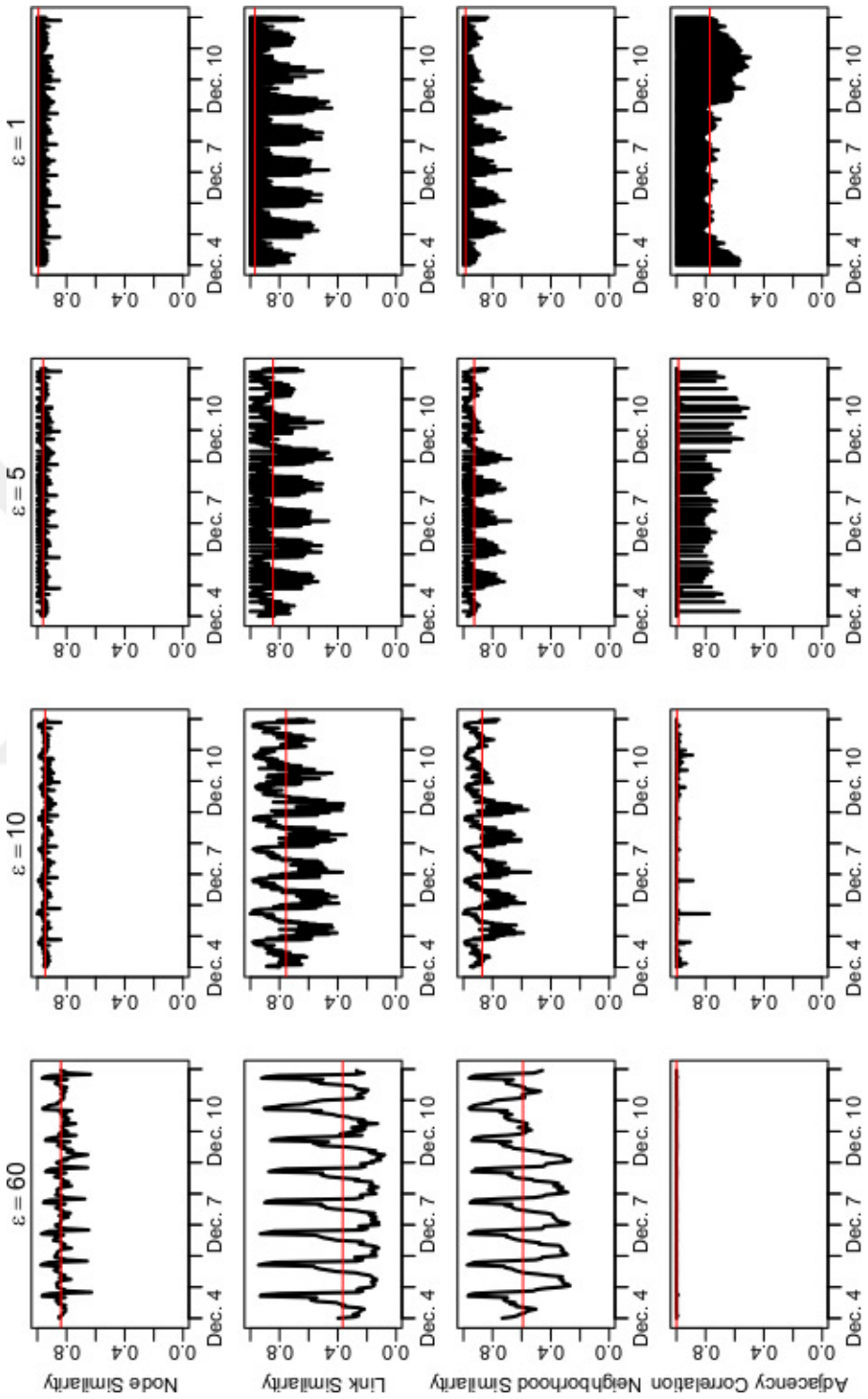


Figure 6.6: Time Series of similarity metrics and γ for the snapshots of $\epsilon = \{1, 5, 10, 60\}$ minutes.

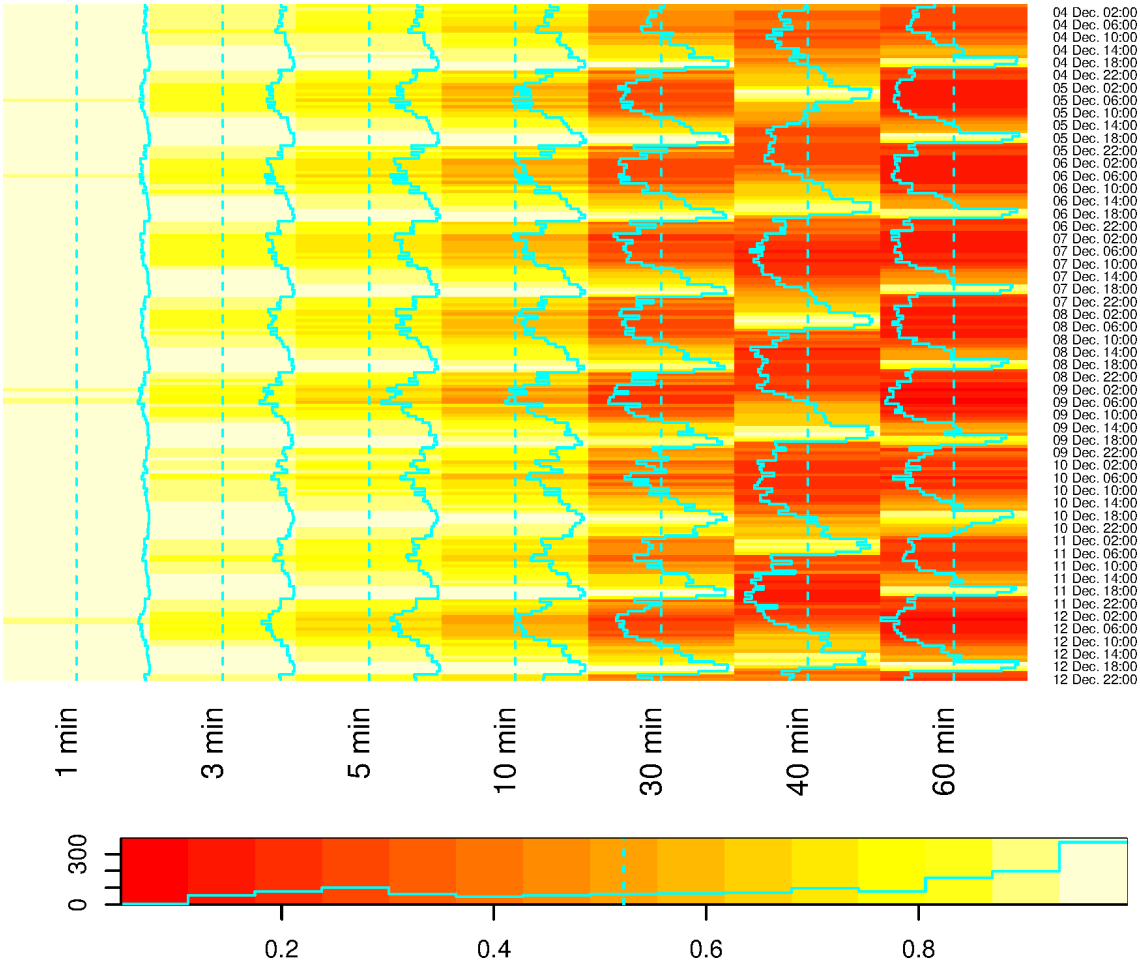


Figure 6.7: Heat map for one week. The x-axis represents the corresponding ϵ and the y-axis represents S_{link} scores in color from red to yellow. The lighter the color, the closer the S_{link} to one.

6.3 Proper Time Interval for WAP Snapshots

The limit intervals of the node and link similarities are shown in figures 6.8 and 6.10 respectively. We have calculated the expected value of the null model for each consecutive snapshots by using equation 5.5. We remind that this score is the expected value of the Jaccard similarity of two consecutive snapshots' node sets and link sets separately. We also consider 0.05 error rate for each value. Hence, the expected values are indicated as the red hatched zone in the plots. If the similarity signal coincides with or above the red hatched zone, than the examined consecutive snapshots stay stable in terms of the studied metric. It means that no need to create a new snapshot, but an aggregation of snapshots is possible. We will concentrate on the signal parts that are out of the red hatched zones when interpreting figures

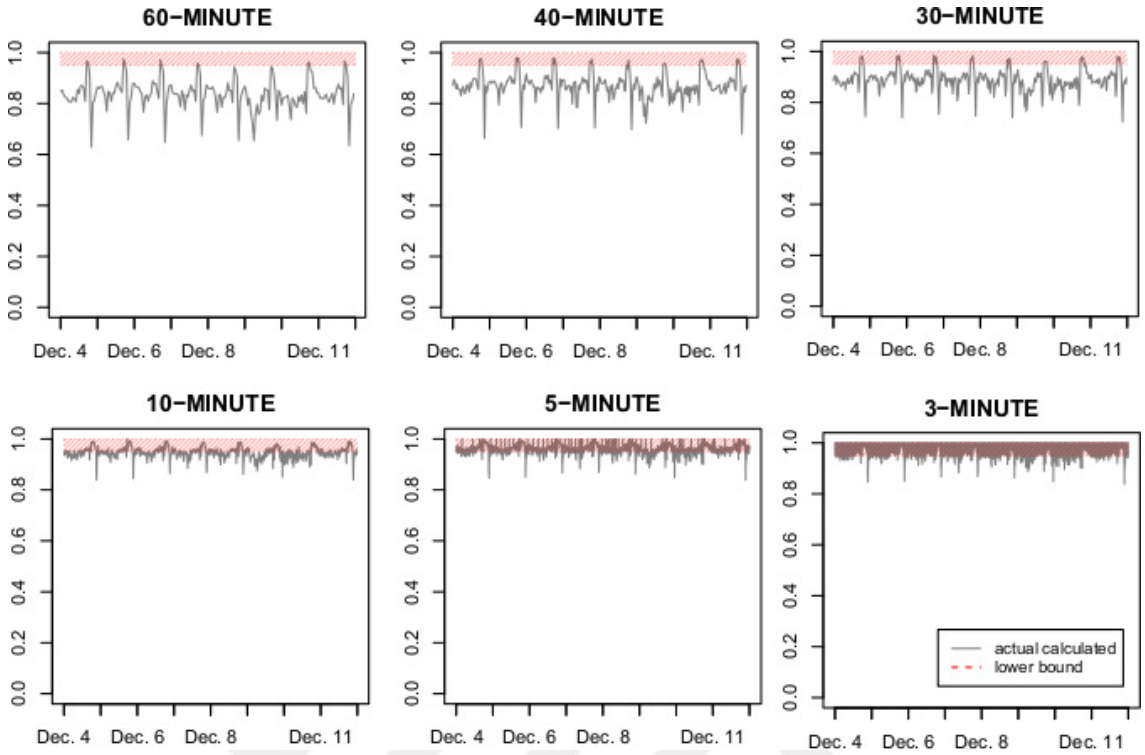


Figure 6.8: Node Similarity Results for different ϵ. Red hatched field is the score which is calculated for null model. We hatched the area between the limit calculated by equation 5.5 and minus 0.05.

6.8 and 6.10.

For S_{node}, when $\epsilon < 10$, signal stays completely inside of the red hatched zone. When ϵ is between 30 and 60 minutes, it meets with a red hatched zone only between 16 : 00 and 20 : 00. For $\epsilon = 10$, one can observe in detail the signal and its expected values in the top plot of figure 6.9. Accordingly, from 02 : 00 at night till 13 : 00 in the afternoon and around 21 : 00 in the evening, S_{node} stays out of the red hatched zone. Thus, new snapshots can be extracted for the system representation at those periods. The other time intervals look like calm periods of the campus with fewer actor change. Hence, snapshots can be aggregated at them.

Regarding S_{link}, mostly, we obtain similar results with S_{node}. Nevertheless, there are also some important different facts related to S_{link}. First, we find out that S_{link} signal meets with or stays outside of the red hatched zone partially, not only larger than 10 but all ϵ. Second, for $\epsilon > 10$ minutes, for some time intervals the expected value is $1.0(-0.05)$ and for other time intervals it is 0.00. This fact means that almost

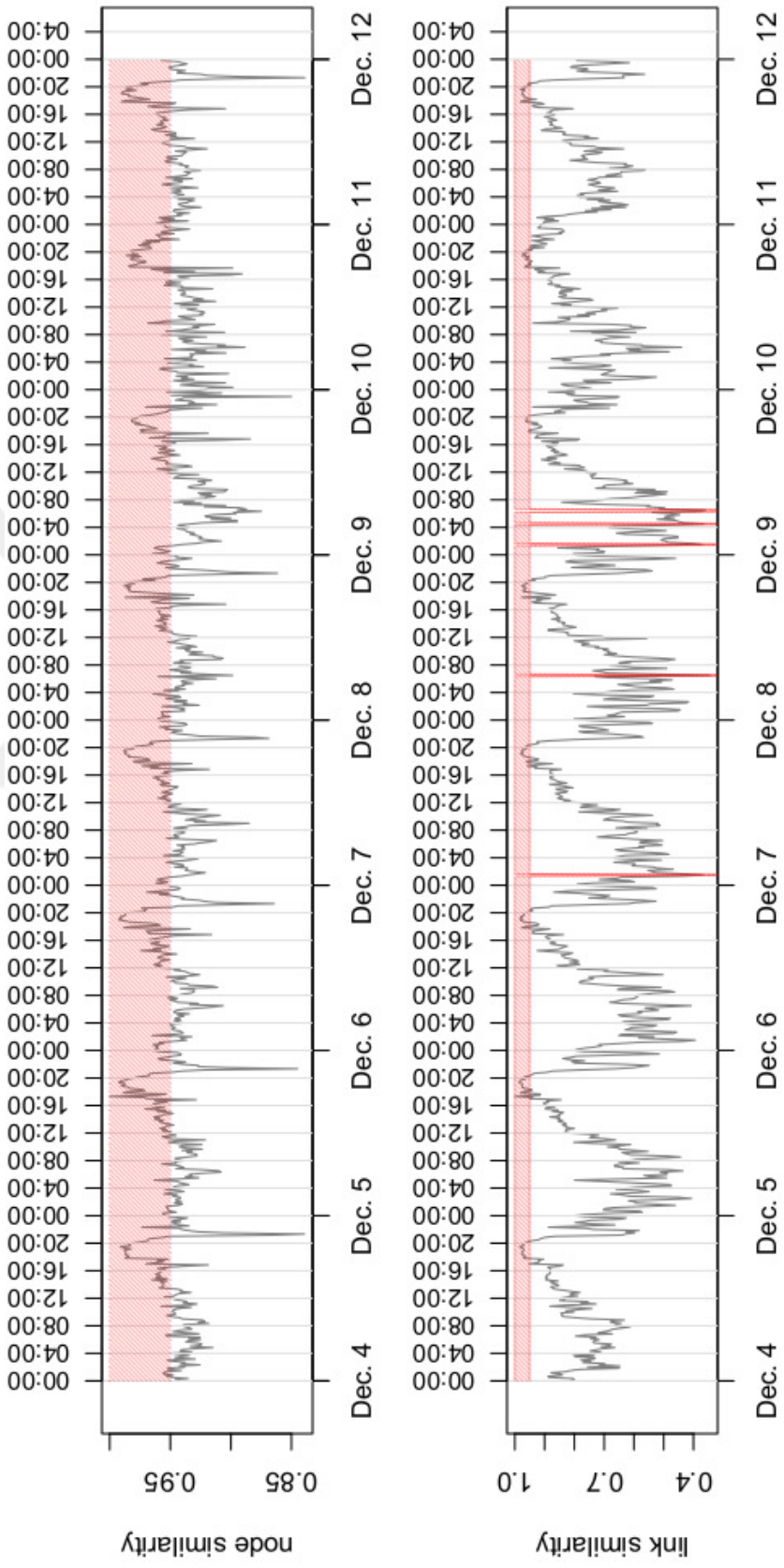


Figure 6.9: Node and link Similarity results for $\epsilon = 10$ minutes. Red hatched field is the score which is calculated for null model. We hatched the area between the limit calculated by equation 5.5 and minus 0.05.

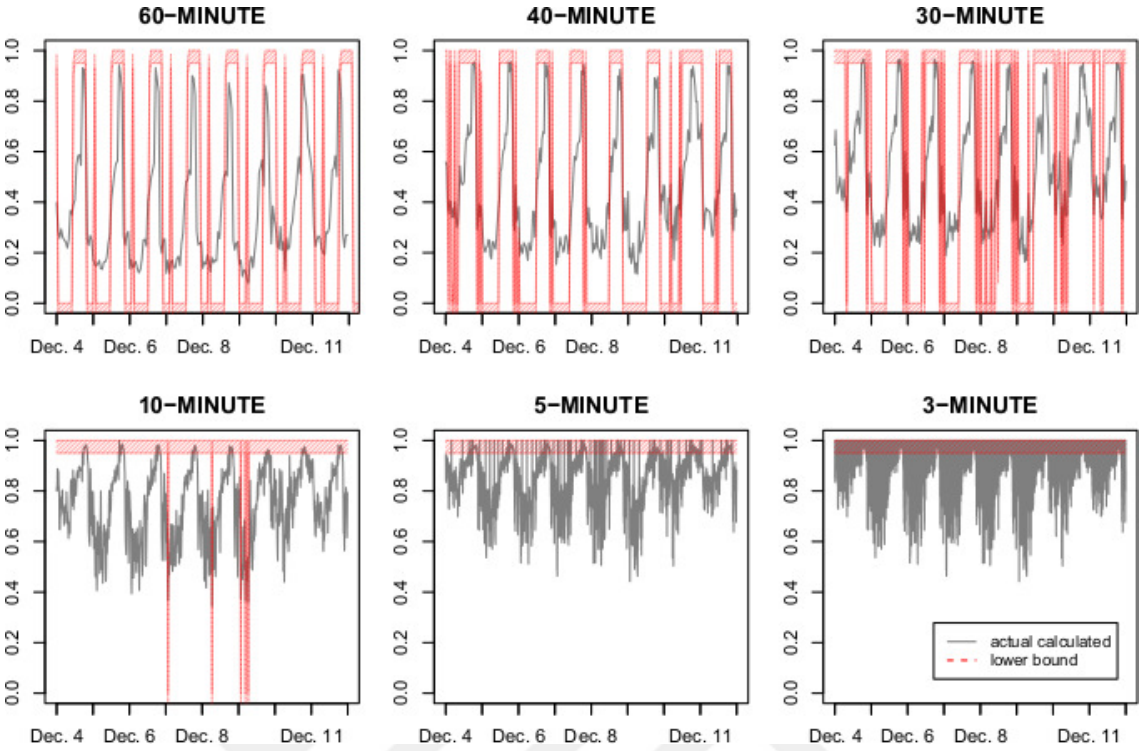


Figure 6.10: Link Similarity Results for different ϵ. Red hatched field is the score which is calculated for null model. We hatched the area between the limit calculated by equation 5.5 and minus 0.05.

no link change in the system is a sign of system stability for some intervals, while for some other intervals, the system is considered stable even if the link changes at a very large rate. When the expected value is close to 0.00, it seems like the continuity of the system in terms of its link structure is lost. In the previous section, we have seen that, in general, S_{link} scores were lower than S_{node} scores. In other words, the system actors' connections change much more than the system actors in consecutive time periods. Interpreting figure 6.10, we notice that when $\epsilon > 10$, we cannot follow the continuity of the system. However, different from S_{node} interpretation, this time, for any $\epsilon \leq 10$, we can aggregate the ϵ for the time periods staying inside of red hatched areas in figure 6.10. Nevertheless, considering topological properties results, similarity results, and the zones computed for the null model altogether, the most appropriate value for ϵ looks like 10 minutes for aggregation. Lower values seem to cause of noisy signals.

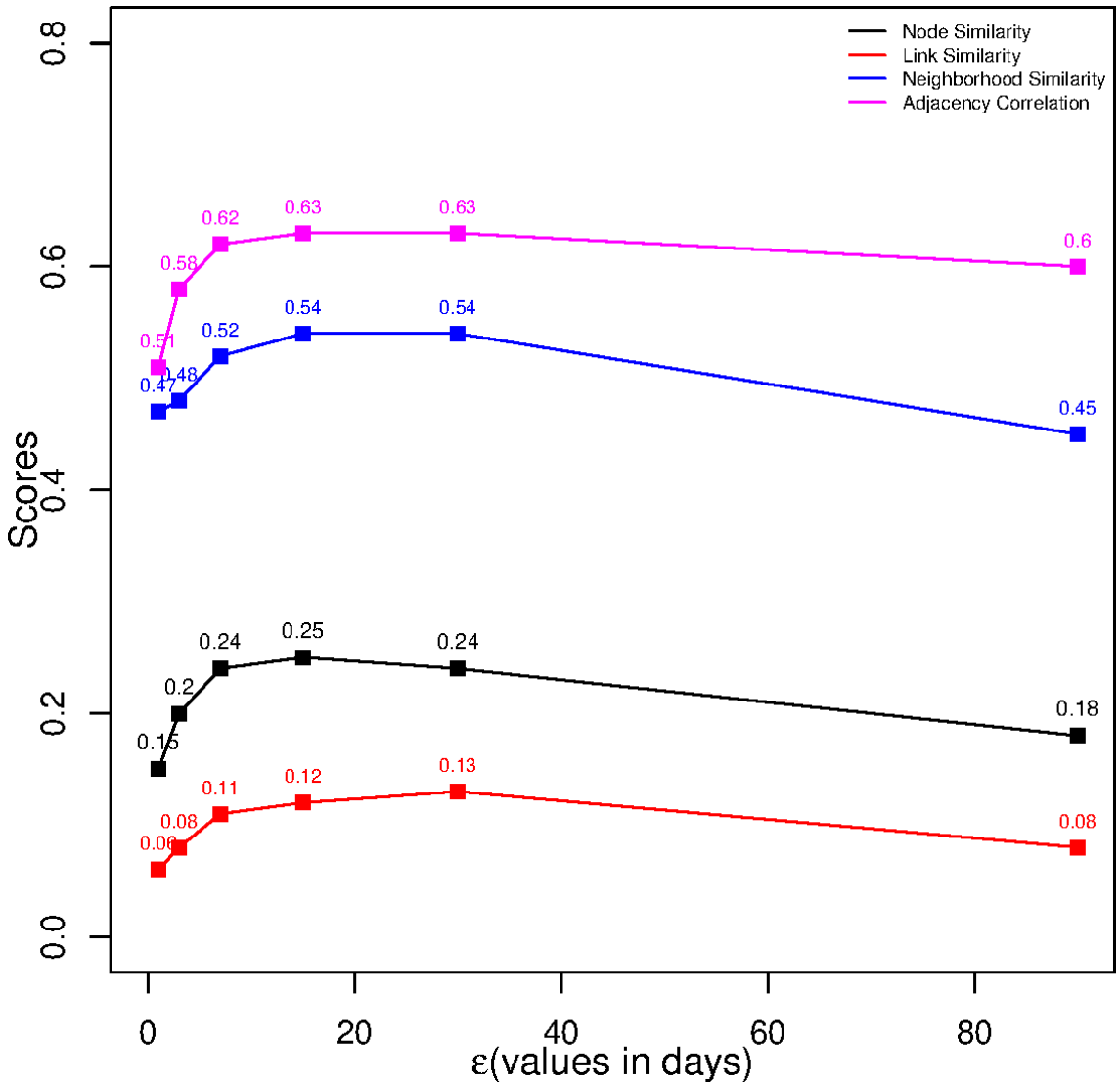


Figure 6.11: Changes on average similarity scores and γ with respect to ϵ.

6.4 Enron Results

Average values and the signal results of similarity metrics and γ that are generated for different ϵ is shown in figure 6.11 and 6.12 respectively. These signal characteristics are very different from the ones of WAP snapshots. The main reasons for these differences are; first, the data characteristics; second, ϵ time scale and third, the evaluation period. We observe a long-range variation in Enron, more than three years, while it was one week for WAP snapshots. In general, the result values are very low compared to WAP snapshots. Average similarity for both proposed metrics and γ increases until ϵ becomes 7 days then declines very slowly. When we look at the details, we see that S_{link} and S_{node} shows a signal like an arc. They first increase

in all ϵ, then remain constant for a long time and then decrease. The clearest plot of this behavior can be seen in figure 6.12 for $\epsilon = 90$. When we investigate the reasons of this, we observe that the snapshots that occur during the time periods when the similarities increase and decrease (the first quarters of 98 and 2002 respectively) are still in the process of forming a robust network. In other words, the node and link structure of snapshots vary completely at these intervals. These are the periods when the network gets denser to build a stable structure.

Regarding γ, the most prominent points are; first, its scores are much higher than the proposed metrics. This means that snapshots are seemed to be more similar from the perspective of γ. Second, it shows an unsteady signal until May 2000. Then the values are decreasing very slowly. For $S_{neighbor}$, the signals seem unsteady. It is both sensitive to ϵ change and also time flow. It seems that the Enron snapshots vary a lot, unlike the WAP example. Even for the smallest ϵ chosen, the scores are too low, indicating that consecutive snapshots are not alike. When trends of change are examined only, there is no informational pattern for determining time intervals. Finding proper time intervals just by analyzing the signals does not seem possible in this example. We should examine whether these signals show the significant similarity of consecutive snapshots. Using S_{link} and S_{node} compared to their expected values helps us a lot in the Enron example.

When we try to find the proper time interval, we encounter similar findings for both S_{node} and S_{link} in the results we obtained by expected value. Accordingly, when $\epsilon = 90$, for all snapshots, even if the similarities are very low, it is higher than the expected value. Thus, this time span is too large to capture the significant change in the system. When $\epsilon > 1$, the similarity results are lower than their expected values in the unstable parts of the system, i.e., at the beginning and end of the time span. That is, the system shows a significant change. When $\epsilon = 1$, both S_{link} and S_{node} results are lower than the expected values for many days. Thus, the system shows a significant change. In figure 6.13, we show the S_{link} and S_{node} trends for the period just before and after Enron announced its bankruptcy. Although both signals fluctuate continuously in the entire time span, it is seen that the change in some days after February 2002 is more than expected; that is, the system has become dynamic. For days when the signals are below the red hatched area, the network can be cut. Other parts can be aggregated.

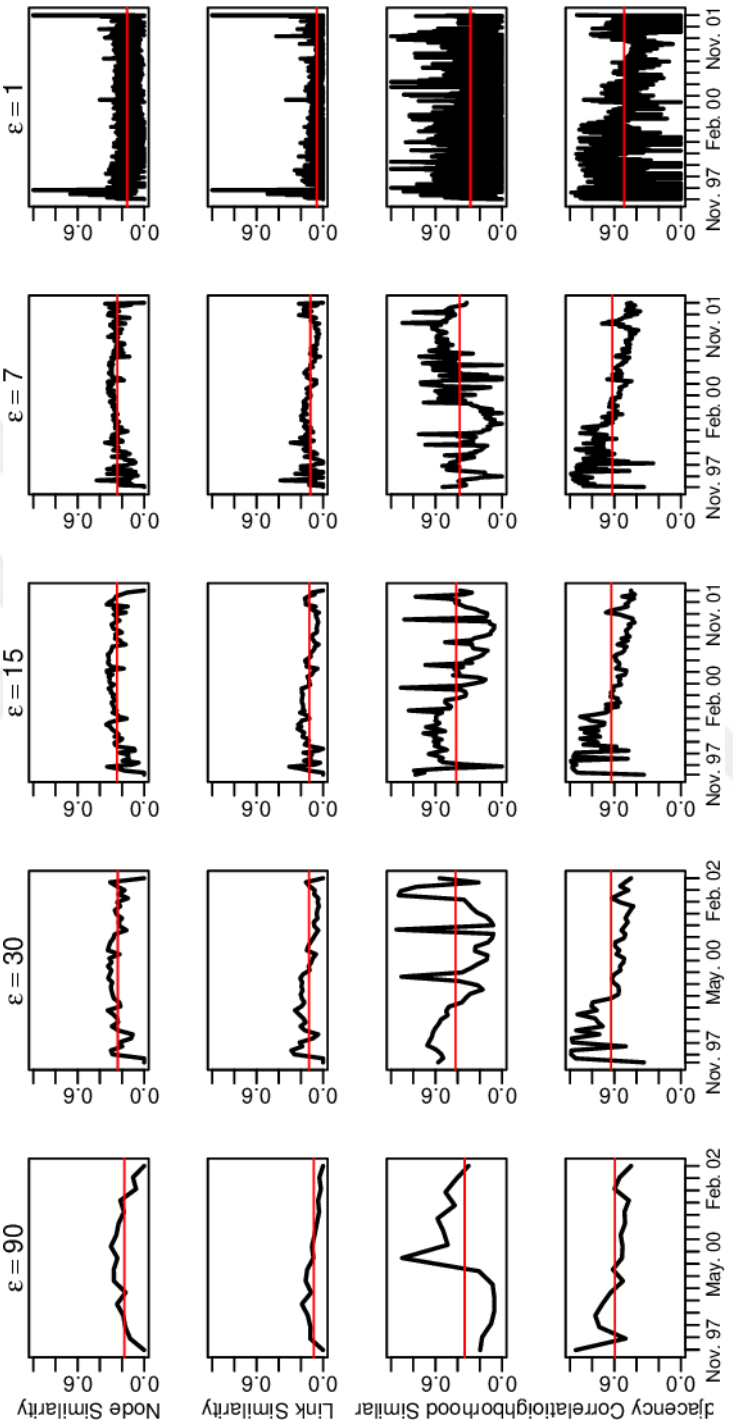


Figure 6.12: Time Series of four similarity metrics for the snapshots of $\epsilon = \{1, 7, 15, 30, 90\}$ days.

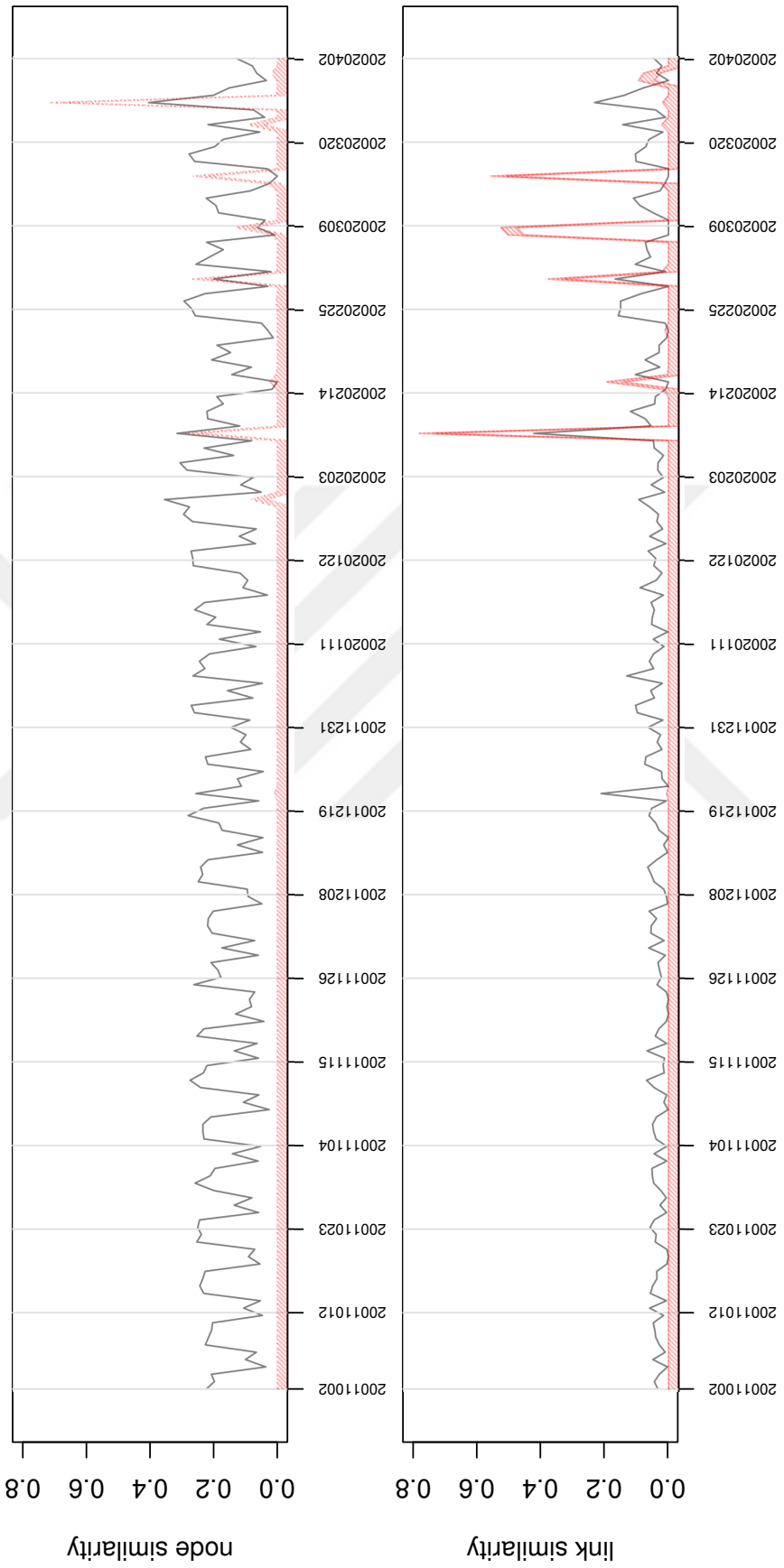


Figure 6.13: Node and link Similarity results for $\epsilon = 1$ day. Red hatched field is the score which is calculated for null model. We hatched the area between the limit calculated by equation 5.5 and minus 0.05.

We do experiments on two different systems that have different temporal dynamics. One of these systems shows a circadian rhythm. It represents the life cycle on campus. The other system spreads over a long period of time. It is the emailing network of Enron. It is constantly changing but not showing a trend of change. The most noticeable results for both experiments can be listed as follows; First, for both systems, the metrics we propose can be used for determining proper time intervals for dynamic network extraction. They are more informative than the adjacency correlation coefficient. Although the two systems we work with have very different properties, the proposed metrics can measure the system's temporal dynamism more effectively, regardless of the studied data. Second, the proposed metrics' statistically expected values can be used to determine the time cut-off points of the entire time span. We come across that usual user-defined thresholds are used in order to decide the significance of similarity metrics. This causes system-specific features to be ignored. In our results, we see that the scores that could be considered low for Jaccard similarity can be higher than their expected values. It allows us to make a more objective evaluation, which is independent than the studied data set. Third, we can reach important campus life findings when the WAP network, which is a network that has not been studied before, is modeled and analyzed as a dynamic network. Our aim here is not to make a detailed data analysis. However, if the WAP network would be analyzed together with the WAP location information, essential findings of life on campus can be obtained. Here we only used this data set to validate the metrics we suggested.

7 CONCLUSION

In this work, we propose a formal definition of dynamic network model extraction independent from the data or analysis method used. This definition is based on the idea that a proper model extraction should contain all necessary snapshots but not more. More clearly, a new snapshot should be created only if there is a critical change. To this end, we concentrated on finding proper discrete sub-interval divisions in which the system stays stable with no critical change. We proposed shifting a window with a very small size, ϵ, and tracking system stability. Indeed, one of the critical points in this approach is measuring the system stability. We proposed three new network similarity metrics; node similarity, link similarity, and neighborhood similarity. These metrics are different versions of Jaccard Similarity to quantify the similarities for consecutive snapshots. We proposed using a null model of those similarities to decide whether the result scores are significant. The values calculated for the null model give us a method independent objective criterion for decision. We empirically tested the usability of proposed similarity metrics for modelling wireless access points log data of the campus of Sabanci University, Istanbul, and Enron Company email data. The former is an original data set that is analyzed for the first time in this work, while the latter is well-known and previously used for the same problem. We extracted different dynamic networks for different ϵ and revealed the effect of ϵ on both proposed similarities and adjacency correlation coefficient, which is another metric from the literature. As a result, we found out that the three similarity metrics are more informative than the adjacency correlation coefficient for both systems. We also caught and interpreted the critical time points when similarities are lower than their expected values for both systems without using any other specific method.

The work presented here can be extended to different paths. First, we presented an empirical study for finding the most appropriate ϵ among the experimental set of ϵ. We compared consecutive ϵ size snapshots and started a new stable interval

if the change between the actual and previous units are large enough. However, if the network is changing slowly, such comparison can be meaningless. In this case, there is no large difference between consecutive snapshots, but as time goes by, the further snapshots apart would be quite different. A more self-working system can be built to automate decision-making and overcome the mentioned problem. For example, an iterative method that automatically shifts the time window and aggregates the snapshots whose similarities are higher than their expected values can be an extension of this work. Thereby slowly changing systems' time span can also be cut off correctly because the comparison would be made on a previously aggregated snapshot with a new ϵ sized snapshot. Second, we compared the similarity metrics with only the adjacency correlation coefficient. It is possible to use other comparison metrics such as adjacency spectral distance to enrich our work. Third, we worked with the time series that are generated either from similarity metrics. We evaluated their graphics and made inferences accordingly. However, signal processing or regression analysis can be complementary to this work. For instance, a signal processing methodology can be performed to discover how much noisy the time series of similarity metrics. Thereby, the proper time interval can be chosen as the least noisy but most informative interval. Such a complete analysis can especially be useful for revealing system facts. In this work, we did not concentrate on data analysis. However, our results on the WAP set show that this type of data contains very rich information about the area's life cycle. Examining such data with a dynamic network perspective can result in rich studies.

REFERENCES

Aggarwal, C., & Subbian, K. (2014). Evolutionary network analysis: A survey. *ACM Computing Surveys*, *47*(1), 10:1–10:36.

Aledavood, T., López, E., Roberts, S. G. B., Reed-Tsochas, F., Moro, E., Dunbar, R. I. M., & Saramäki, J. (2015). Daily rhythms in mobile telephone communication. *PLOS ONE*, *10*(9), 1–14.
URL `https://doi.org/10.1371/journal.pone.0138098`

Alessandrini, A., Gioia, C., Sermi, F., Sofos, I., Tarchi, D., & Vespe, M. (2017). Wifi positioning and big data to monitor flows of people on a wide scale. In *2017 European Navigation Conference (ENC)*, (pp. 322–328).

Berger-Wolf, T. Y., & Saia, J. (2006). A framework for analysis of dynamic social networks. In *Proceedings of the 12th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*, KDD '06, (pp. 523–528). New York, NY, USA: ACM.

Blonder, B., Wey, T. W., Dornhaus, A., James, R., & Sih, A. (2012). Temporal dynamics and network analysis. *Methods in Ecology and Evolution*, *3*(6), 958–972.

Caceres, R. S., Berger-Wolf, T., & Grossman, R. (2011). Temporal scale of processes in dynamic networks. In *2011 IEEE 11th International Conference on Data Mining Workshops*, (pp. 925–932).

Clauset, A., & Eagle, N. (2012). Persistence and periodicity in a dynamic proximity network. *CoRR*, *abs/1211.7343*.

Dakiche, N., Tayeb, F. B., Slimani, Y., & Benatchba, K. (2018). Sensitive analysis of timeframe type and size impact on community evolution prediction. In *2018 IEEE International Conference on Fuzzy Systems*, (pp. 1–8).

Eagle, N., & (Sandy) Pentland, A. (2006). Reality mining: Sensing complex social systems. *Personal Ubiquitous Comput.*, *10*(4), 255–268.

Fish, B., & Caceres, R. S. (2015). Handling oversampling in dynamic networks using link prediction. In *Machine Learning and Knowledge Discovery in Databases*, (pp. 671–686). Cham: Springer International Publishing.

Fish, B., & Caceres, R. S. (2017). A supervised approach to time scale detection in dynamic networks. *CoRR*, *abs/1702.07752*.
URL http://arxiv.org/abs/1702.07752

Holme, P., & Saramäki, J. (2012). Temporal networks. *Physics Reports*, *519*(3), 97–125.

Jaccard, P. (1912). The distribution of the flora in the alpine zone. *New Phytologist*, *11*(2), 37–50.
URL http://www.jstor.org/stable/2427226?seq=3

Jo, H.-H., Karsai, M., Kertész, J., & Kaski, K. (2012). Circadian pattern and burstiness in mobile phone communication. *New Journal of Physics*, *14*(1), 013055.
URL https://doi.org/10.1088%2F1367-2630%2F14%2F1%2F013055

K. Darst, R., Granell, C., Arenas, A., Gomez, S., Saramäki, J., & Fortunato, S. (2016). Detection of timescales in evolving complex systems. *Scientific Reports*, *6*.

Kempe, D., Kleinberg, J., & Kumar, A. (2000). Connectivity and inference problems for temporal networks. In *Proceedings of the Thirty-second Annual ACM Symposium on Theory of Computing*, STOC '00, (pp. 504–513). New York, NY, USA: ACM.

Kjærgaard, M. B., & Nurmi, P. (2012). Challenges for social sensing using wifi signals. In *Proceedings of the 1st ACM Workshop on Mobile Systems for Computational Social Science*, MCSS '12, (pp. 17–21). New York, NY, USA: ACM.

Krings, G., Karsai, M., Bernhardsson, S., Blondel, V. D., & Saramäki, J. (2012). Effects of time window size and placement on the structure of an aggregated communication network. *EPJ Data Science*, *1*(1), 4.

Kuhn, F., & Oshman, R. (2011). Dynamic networks: Models and algorithms. *ACM SIGACT News*, *42*(1), 82–96.

Leskovec, J., Adamic, L. A., & Huberman, B. A. (2007). The dynamics of viral marketing. *ACM Trans. Web*, *1*(1).

Leskovec, J., Kleinberg, J., & Faloutsos, C. (2005). Graphs over time: Densification laws, shrinking diameters and possible explanations. In *Proceedings of the Eleventh ACM SIGKDD International Conference on Knowledge Discovery in Data Mining*, KDD '05, (pp. 177–187). New York, NY, USA: ACM.

Medo, M., Zeng, A., Zhang, Y., & Mariani, M. S. (2018). Optimal timescale of community detection in growing networks. *CoRR*, *abs/1809.04943*.

Real, R. (1999). Tables of significant values of jaccard's index of similarity. *Miscellania Zoologica*, *22*(1), 29–40.

Real, R., & Vargas, J. M. (1996). The probabilistic basis of jaccard's index of similarity. *Systematic Biology, JSTOR*, *45*(3), 380–385.
URL `www.jstor.org/stable/2413572`

Rocha, L. E. C., Masuda, N., & Holme, P. (2017). Sampling of temporal networks: Methods and biases. *Phys. Rev. E*, *96*, 052302.
URL `https://link.aps.org/doi/10.1103/PhysRevE.96.052302`

Rodriguez-Lozano, D., Gomez-Pulido, J. A., & Duran-Dominguez, A. (2016). Predicting access points workload in wi-fi infrastructures according to users' behavior. In *Proceedings of the International Conference on Big Data and Advanced Wireless Technologies*, BDAW '16, (pp. 21:1–21:6). New York, NY, USA: ACM.

Rossi, R. A., Gallagher, B., Neville, J., & Henderson, K. (2013). Modeling dynamic behavior in large evolving graphs. In *Proceedings of the Sixth ACM International Conference on Web Search and Data Mining*, WSDM '13, (pp. 667–676). New York, NY, USA: ACM.

Schieber, T. A., Carpi, L., Díaz-Guilera, A., Pardalos, P. M., Masoller, C., & Ravetti, M. G. (2017). Quantification of network structural dissimilarities. *Nature Communications*, *8*, 13928 EP –. Article.
URL `https://doi.org/10.1038/ncomms13928`

Shimada, Y., Hirata, Y., Ikeguchi, T., & Aihara, K. (2016). Graph distance for complex networks. *Scientific Reports*, *6*, 34944. Article.
URL `https://doi.org/10.1038/srep34944`

Song, C., Qu, Z., Blumm, N., & Barabási, A.-L. (2010). *Science*, *327*(5968), 1018–1021.

Soundarajan, S., Tamersoy, A., Khalil, E. B., Eliassi-Rad, T., Chau, D. H., Gallagher, B., & Roundy, K. (2016). Generating graph snapshots from streaming edge data. In *Proceedings of the 25th International Conference Companion on World Wide Web*, WWW '16 Companion, (pp. 109–110). Republic and Canton of Geneva, Switzerland: International World Wide Web Conferences Steering Committee.
URL `https://doi.org/10.1145/2872518.2889398`

Spiliopoulou, M. (2011). Evolution in social networks: A survey. In *Social Network Data Analytics*, chap. 6, (pp. 149–175). Springer.

Stopczynski, A., Sekara, V., Sapiezynski, P., Cuttone, A., Madsen, M. M., Larsen, J. E., & Lehmann, S. (2014). Measuring large-scale social networks with high resolution. *PLOS ONE*, *9*(4), 1–24.

Sulo, R., Berger-Wolf, T., & Grossman, R. (2010). Meaningful selection of temporal resolution for dynamic networks. In *Proceedings of the Eighth Workshop on Mining and Learning with Graphs*, MLG '10, (pp. 127–136). New York, NY, USA: ACM.

Uddin, S., Choudhury, N., M. Farhad, S., & Rahman, M. (2017). The optimal window size for analyzing longitudinal networks. *Scientific Reports*, *7*.

Wills, P., & Meyer, F. (2019). Metrics for graph comparison: A practitioner's guide. *CoRR*.

Zhang, N., Yang, P., Ren, J., Chen, D., Yu, L., & Shen, X. (2018). Synergy of big data and 5g wireless networks: Opportunities, approaches, and challenges. *IEEE Wireless Communications*, *25*(1), 12–18.

Zhang, Y., Wang, L., Zhang, Y.-Q., & Li, X. (2012). Toward a temporal network analysis of interactive wifi users. *EPL (Europhysics Letters)*, *98*, 68002.